

Improving Performance of Router-Assisted Transport Protocols over Variable Capacity Links

Ihsan Ayyub Qazi

Department of Computer Science

LUMS School of Science and Engineering, Lahore, Pakistan

Email: ihsan.qazi@lums.edu.pk

Taieb Znati and Daniel Mosse

Department of Computer Science

University of Pittsburgh, Pittsburgh, PA USA

Email: {znati,mosse}@cs.pitt.edu

Abstract—Many promising congestion control protocols use explicit feedback from the network to achieve high performance. These protocols often use congestion signals whose computation requires an estimate of link capacity. Such estimates are not available in networks where capacity varies over time. This paper studies the impact of inaccurate capacity estimates on the performance of congestion control protocols over variable capacity links. As a case study, we focus on 802.11 WLANs. We show that such estimates can lead to either under-utilization or unfairness and network overload. Using a model, we characterize the available capacity of a node in a 802.11 WLAN and then study a method for capacity estimation. Using simulations, we show that the method leads to high utilization and fairness over shared, multi-access networks.

I. INTRODUCTION

Technological advances in the last few decades have led to the deployment of very high speed links on the Internet. Proliferation of wireless networks coupled with the increased diversity of applications on the Internet has stressed the congestion control algorithm in the Transmission Control Protocol (TCP) [1]. To meet these challenges, researchers have proposed several congestion control protocols that use explicit feedback from the network [1], [2], [3].

Many of these protocols (e.g., XCP [1], RCP [2], BMCC [3]) use feedback signals that provide a more accurate characterization of congestion than signals used by various versions of TCP, such as packet loss and delay. Improved feedback allows these protocols to achieve efficient and fair bandwidth allocations while maintaining low queues and negligible loss rates. However, computation of such feedback often requires an estimate of capacity (see Table I). While capacity estimates are available in wired point-to-point links, this is not the case in shared, multi-access networks, such as 802.11 Wireless LANs (WLAN), where capacity changes over time. The absence of accurate capacity estimates poses a significant deployment challenge for such protocols [4].

In this paper, we ask, “*What is the impact of inaccurate capacity estimates on the performance of congestion control protocols that require them for feedback computation?*” Our results show that inaccurate estimates can lead to either under-utilization or unfairness and network overload. To address this, we study the use of per-packet throughput observed at the network layer as an estimate of capacity. Our evaluation focusses on 802.11 WLANs since most end-to-end paths

Protocol	Feedback	Equations Involving Capacity Estimate
RCP [2]	per-flow rate	$R(t - t_p)[1 + \frac{t_p}{d}(\frac{\alpha(C - y(t)) - \beta \frac{q(t)}{d}}{C})]$
XCP [1]	per-flow $\Delta cwnd$	$\alpha d(y(t) - C) - \beta q$
BMCC [3]	load factor	$(\lambda + \kappa_1 q) / \gamma C t_p$
MaxNet [5]	congestion price	$p(t) + t_p \cdot (y(t) - \gamma C) / C$

TABLE I

FEEDBACK USED BY DIFFERENT PROTOCOLS WHERE C , R , y , λ , q , p , d , t_p , AND γ IS THE LINK CAPACITY, FLOW RATE, ARRIVAL RATE, ARRIVED BYTES, QUEUE LENGTH, ROUND-TRIP TIME, UPDATE INTERVAL, AND TARGET UTILIZATION, RESPECTIVELY AND THE REST ARE CONSTANTS.

today contain a wireless segment. Moreover, we consider scenarios where the wireless segment is the bottleneck, which is increasingly the case in Internet cafes, offices, and airports.

Our results show that per-packet throughput estimates can vary widely over short timescales due to variations in channel conditions. To reduce the noise in estimates, capacity is measured over an interval, whose value presents a tradeoff between estimation accuracy and congestion responsiveness. We show that exponential averaging across intervals can help in addressing this tradeoff. Using simulations, we then evaluate the impact of resulting estimates on the performance of the recently proposed Binary Marking Congestion Control (BMCC) protocol [3]. Our results show that these estimates allow BMCC to achieve high throughput and fairness in diverse scenarios. However, under high channel loss rates, performance can still degrade due to decrease in estimation accuracy (e.g., 50% loss rate introduced up to 20% estimation error). Lower MAC bitrates result in smaller errors as they are more robust to losses. When clients have heterogeneous channel conditions to the Base Station (BS), these estimates lead to throughput proportional to the loss rate experienced by each client. This suggests that these estimates can be adapted to provide service differentiation to applications.

The rest of the paper is organized as follows: We discuss related work in Section II and study the impact of inaccurate capacity estimates in Section III. In Section IV, we present a model to characterize the capacity of a node in a 802.11 WLAN. We study a method for capacity estimation in Section V, evaluate its performance with BMCC in Section VI, and offer concluding remarks in Section VII.

II. RELATED WORK

Several congestion control protocols have been proposed that rely on capacity estimates for feedback computation.

XCP [1] and RCP [2] use these estimates to determine per-flow window changes and rates, respectively, MaxNet [5] for computing the congestion price and BMCC [3] and MLCP [6] for determining the load factor. Some of these protocols have been considered in wireless settings, for instance, performance of XCP in wireless environments was studied in [4].

In [4], queue variation/speed is used for estimating spare bandwidth. This approach does not provide estimates when the queue is empty. Moreover, it can result in widely varying estimates because changes in queue occupancy occur not only due to variations in channel conditions but also due to bursty packet arrivals. Furthermore, the proposed algorithm is closely tied with XCP and thus difficult to generalize. In [7], the output traffic rate is used as an estimate for capacity. While this approach can provide good estimates when a node always has a packet to send, it can lead to estimation errors when the input traffic rate is less than the available capacity.

In [8], authors use analytical models of the 802.11 MAC to estimate capacity. While such approaches can provide accurate estimates, they can be hard to use in general settings due to model limitations. In [9], [10], [11], [12], either inter-packet delays or packet transmission delays are used for capacity estimation. The usage of measured throughput of packets for bandwidth estimation has been employed in [9], [10], [11] but in different contexts. In [11], these estimates are used in a dynamic admission control scheme for providing soft rate guarantees in WLANs whereas [10] uses them to improve the performance of adaptive multimedia applications in AODV MANETs. However, such estimates have not been used in a congestion control protocol, such as BMCC, in which flows adapt their rates based on network feedback.

III. IMPACT OF INACCURATE CAPACITY ESTIMATES

In this section, we study the impact of capacity estimates on the performance of BMCC over a 802.11 WLAN using ns2 simulations. We first provide a brief background on BMCC.

A. Background: BMCC

With BMCC [3], each router computes the load factor (ratio of demand to capacity) every t_p interval as shown in Table I. This is explicitly communicated to the sources that apply either multiplicative increase, additive increase, or the multiplicative decrease policy depending on the bottleneck load factor. BMCC achieves efficient and fair bandwidth allocations on large bandwidth-delay product (BDP) networks while maintaining low bottleneck queue and loss rate. We use BMCC in our evaluation for two reasons: (i) it outperforms TCP SACK, XCP, and in some cases RCP, in terms of flow completion times for typical Internet flow sizes and (ii) it is particularly amenable to deployment as it uses only the existing ECN bits for representing congestion feedback.

B. Evaluating the Impact of Capacity Estimates

We consider a 802.11b WLAN with five clients. The modulation rate is fixed at 11 Mbps and the channel conditions between the clients and the BS are assumed to be similar. We

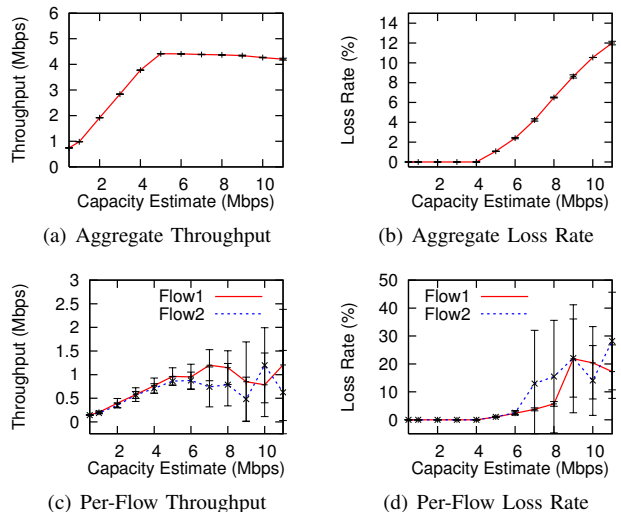


Fig. 1. Performance of BMCC clients as a function of the capacity estimate.

start a long-lived download flow on each client. We vary the downlink capacity estimate, C_e , from the BS to the clients and measure the throughput and loss rate of each client.

Figures 1(a) and 1(b) show the aggregate throughput and loss rate as a function of C_e , respectively. In this scenario, the true downlink capacity, C_l , is ≈ 4.2 Mbps (see Section IV). When $C_e < C_l$, the aggregate throughput increases linearly with C_e and the loss rate remains negligible. Since the estimate is smaller than the true capacity, the BS advertises higher than actual load factor, similar to a virtual queue [13], which causes sources to backoff early resulting in an under-utilization of $(1 - C_e/C_l)$. When $C_e > C_l$, the BS indicates lower than true load, causing sources to send more packets than can be handled by the network, which increases packet losses. Observe that the aggregate loss rate increases almost linearly with C_e in this regime and becomes $\approx 12\%$ when $C_e = 11$ Mbps.

We now analyze the impact of varying capacity estimates on per-flow performance. Figures 1(c) and 1(d) show the throughput and loss rate of two (out of the five) randomly chosen flows, respectively. Observe when $C_e < C_l$, each flow gets equal throughput, experience negligible losses and incur little variations across runs as shown by the error bars. However, when C_e is increased beyond C_l Mbps, flows start experiencing more losses which leads to degraded fairness and unstable throughput behavior (observe the large variations in throughput). With higher capacity wireless MACs, such as 802.11n, performance degradation can be even more significant. *These results show that in order to prevent under-utilization or unfairness and network overload, the time-varying capacity should be estimated accurately.*

IV. LINK CAPACITY

In this section, we present a simple model to study the key factors that affect the capacity of a wireless link. In contrast to existing MAC models (e.g., see [14] and references therein), the model is not specific to a particular MAC and is applicable

to all shared, multi-access networks. In this work, however, we consider its application in 802.11 WLANs.

Let C_l be the *maximum achievable throughput* or capacity above the MAC layer on link l , where a link refers to a sender-receiver pair. Suppose the PHY bitrate used by a node on link l is R_l bps and the channel loss rate is p_l . As the wireless medium is a shared resource, the air time available on link l depends on the air time used by transmissions on other links. Let $U_{S-\{l\}}$ be the fraction of air time utilized by transmissions on links other than link l , where S is the set of all links in a WLAN. Then the PHY layer capacity of link l is given by

$$C_l^{PHY} = R_l(1 - p_l)(1 - U_{S-\{l\}}) \quad (1)$$

To determine C_l , we need to account for all the overheads induced by the 802.11 MAC and PHY. Let the transmission time of a MAC frame on link l be T_l^{tr} and the overhead introduced by the MAC and PHY layers be T_l^{ov} , then

$$C_l = \frac{T_l^{tr}}{T_l^{tr} + T_l^{ov}} \cdot C_l^{PHY} \quad (2)$$

We now characterize T_l^{ov} in a 802.11 WLAN.

802.11 MAC+PHY Overhead: The 802.11 DCF introduces a variable amount of overhead on every packet it transmits [14]. For unicast transmissions, these overheads correspond to the time needed to gain channel access, synchronize transmissions, and transmit a link layer ACK. Therefore, the total time taken by the interface to send a packet is given by $T_l^{tr} + T_l^{ov}$. $T_l^{tr} = X/R_l$ where X is the MAC frame size and

$$T_l^{ov} = \text{DIFS} + T^{pr} + T_l^{BO} + \text{SIFS} + T^{pr} + T^{ack} \quad (3)$$

where T^{pr} is the transmission time of the PLCP preamble and header, T_l^{BO} is the average backoff experienced by packets on link l , T^{ack} is the ACK transmission time, and DIFS and SIFS are the inter-frame spacings. The capacity above the network layer is given by $(X - H/X) \cdot C_l$, where H is the sum of the MAC and IP header sizes. For reliable protocols, such as TCP and BMCC, the overhead of transport layer ACKs also needs to be considered. Using Equation 2, we can now derive the capacity. For instance, consider a 802.11b WLAN, where $X=1500$ b, $R_l=11$ Mbps, and $T_l^{BO}=320 \mu\text{s}$. This gives a BS→client capacity of ≈ 6 Mbps for UDP traffic and ≈ 4.2 Mbps (assuming 40 b transport ACKs) for BMCC.

V. ESTIMATING CAPACITY

In the previous section, we showed that the link capacity depends on the PHY bitrate, channel loss rate, free air time, packet size, and the variable amount of per-packet overhead introduced by the underlying MAC and PHY layers. One method for estimating capacity is to directly use Equation 2 by measuring the average values of p_l , $U_{S-\{l\}}$, R_l , T^{BO} , and X in a given interval T . However, this involves measurement and tracking of several variables by the MAC. A simpler approach is to use the average transmission rate experienced by successful network layer packets as estimates for capacity [10], [11]. This requires maintaining only the average transmission time of packets, which accounts for all possible delays introduced by lower layers. In this work, we take the latter approach.

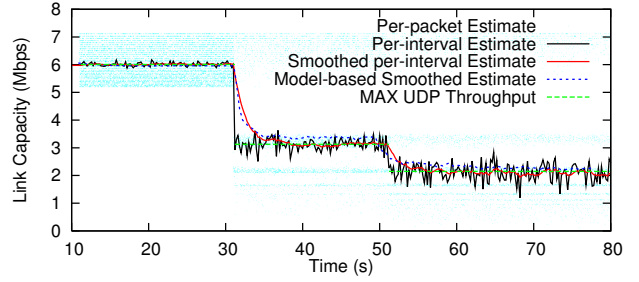


Fig. 2. Compares the maximum UDP throughput with different capacity estimates. The model-based estimates are obtained using Equation 2 and smoothed per-interval estimates are obtained through exponential averaging.

A. Using Packet Transmission Times for Capacity Estimation

The transmission time of a network layer packet is the difference between t_s , the time when a packet is released to the MAC and t_e , when the corresponding MAC layer ACK is received. Note that $t_s - t_e$ includes MAC overhead delays, deferrals due to channel contention, retransmissions due to packet losses, the transmission delay, and the propagation delay. Given a packet of size X bits, the network-layer transmission rate is equal to $X/(t_e - t_s)$.

Capacity Measurement Interval: The wireless channel and the 802.11 MAC can introduce significant variations in these estimates on short timescales (see Figure 2). To reduce these variations, we measure capacity over an interval T . The value of T presents a tradeoff between accuracy and congestion responsiveness and is related to the round-trip time flows. While small T results in noisy estimates, large T reduces responsiveness to congestion. To balance this tradeoff, we set T to the same value as used by network-based congestion control protocols for computing feedback. For instance, XCP and RCP dynamically vary the measurement interval depending on the average round-trip of flows, whereas BMCC uses a fixed value, larger than the average round-trip of flows on the Internet [3]. Note that averaging per-packet transmission rates can bias the estimate towards higher rates. To remove this bias, we measure the total bytes sent in an interval and divide this by the sum of the packet transmission times.

Figure 2 compares the maximum UDP throughput with different capacity estimates for three clients that join the WLAN with an inter-arrival time of 20s. The maximum UDP throughput is obtained by making each client send packets in backlogged mode. Observe that as new clients arrive, the per-interval transmission rate is able to track capacity quite accurately. However, the increase in the number of clients increases the burstiness in per-interval estimates. To reduce variations in estimates, we use exponential averaging across intervals as follows: $C_e^{av} = a \cdot C_e^{av} + (1 - a) \cdot C_e^{sample}$, where C_e^{av} is the running average, C_e^{sample} is the current capacity sample, and a is a filter constant. Note that the model-based estimates (computed using Equation 2) are similar to the transmission rate estimates.

Impact of Channel Losses on Estimation Accuracy: Channel losses can often occur in a wireless medium due to signal

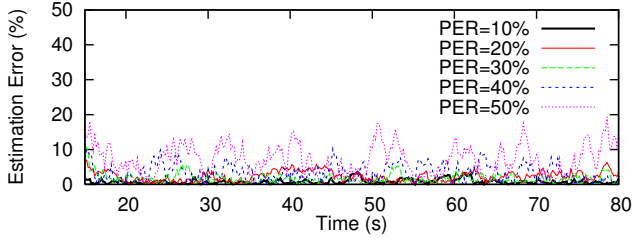


Fig. 3. Capacity estimation error as a function of time for different PERs.

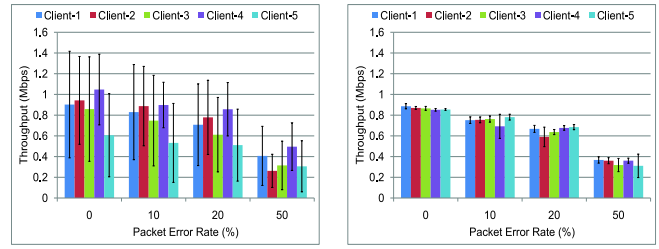
attenuation, fading, and shadowing effects. These losses can affect the accuracy of capacity estimates. Assuming an error model that introduces packet errors uniformly at random, we vary the Packet Error Rate (PER) from 10% to 50% and analyze the capacity estimation error, $|C_e - C_a|/C_a$, where C_a and C_e are the actual and estimated capacities, respectively. The actual capacity is computed using Equation 2.

Figure 3 shows the estimation error as a function of time for different PERs. Observe that for PERs less than 30%, the estimation error remains below 5%. As the PER increases, the estimation error also increases, reaching an average value of $\approx 10\%$ (and a maximum value of 20%) for PER=50%. This happens because as the PER increases, packets start experiencing burst losses that result in successive retransmissions at the MAC layer and an increase in the average backoff period. This increases the variability in packet transmission times, resulting in larger estimation errors.

Handling Heterogenous Packet Sizes: Larger packet sizes result in higher measured throughput as they allow more bytes to be sent on every channel access. In the presence of packets with different sizes, the measured capacity can vary widely. To handle this heterogeneity, we can make capacity estimation independent of the packet size. To achieve this, each wireless node maintains the average size of all the packets seen in a moving window. The available capacity is then determined by normalizing the transmission rate of a packet to the average packet size. Observe that the transmission rate of two packets only differ in their times to transmit a data packet at the link layer. The normalized transmission rate is then given by $X_{av}/(T_X + X_{av}/R_l - X/R_l)$, where X is the packet size, X_{av} is the average packet size, and T_X is the transmission time of a packet with size X . Note that capacity estimation is not affected by link-layer fragmentation.

VI. PERFORMANCE EVALUATION

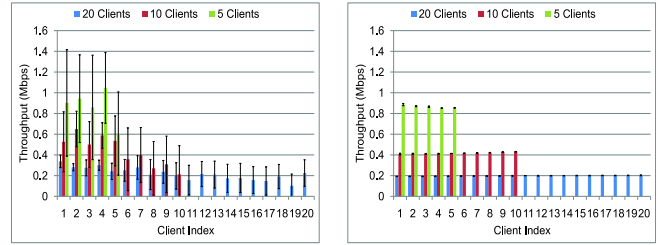
In the last section, we showed that packet transmission rates can be effectively used as estimates of capacity. In this section, we evaluate the impact of using these estimates on the end-to-end performance of BMCC by allowing each node in a WLAN to compute the load factor feedback based on these estimates. We consider diverse network scenarios in our evaluation and study the impact of channel losses, number of clients, PHY bitrates, and heterogeneous Client-BS channel conditions on the performance of BMCC.



(a) Without estimation ($C_e=11$ Mbps)

(b) With estimation

Fig. 4. Throughput of five clients as a function of the channel loss rate



(a) Without estimation ($C_e=11$ Mbps)

(b) With estimation

Fig. 5. Throughput of BMCC as a function of the number of clients.

A. Simulation Setup

All simulations are conducted in ns2 version 2.33, which we have extended with a capacity estimation module at the network layer. We use the default parameters for 802.11b in ns2. The bitrate is fixed at 11 Mbps, unless stated otherwise. All the wireless nodes are assumed to be within the transmission range of each other. We run bulk transfer flows on each client with a data packet size of 1500 b. The interface queue size is set to 64 packets. The capacity measurement interval, T , is set to 200 ms, the same interval over which BMCC computes its feedback, and the filter constant, a , is set to 0.875 as in [6]. All simulations are run for at least 180 s and the results are averaged over 10 runs.

B. Impact of Channel Losses

In the presence of channel losses, network layer packets may experience multiple retransmissions at the MAC layer. This can negatively impact end-to-end performance due to increased variability in estimates, which occurs for two reasons: (1) since losses are random, some packets take longer to complete than others and (2) more losses mean fewer packets get sent in a measurement interval. Figure 4 shows the throughput achieved by five BMCC clients as a function of the PER. Observe that across a range of PERs, using an inaccurate capacity estimate leads to unfairness and large throughput variations. When capacity estimation is used, BMCC clients achieve a fairer throughput distribution. However, larger PERs still lead to noticeable variability in the throughput of clients.

C. Impact of the Number of Clients

When the number of clients in a 802.11 WLAN increases, clients start experiencing more collisions, which causes each

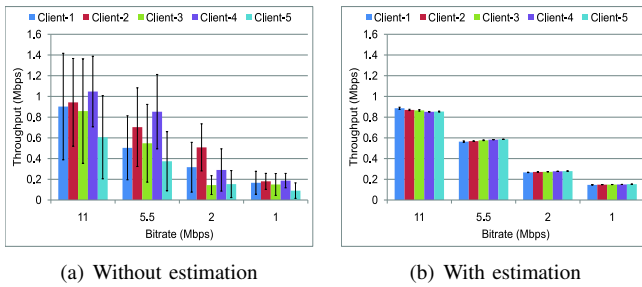


Fig. 6. Throughput of five clients as a function of the PHY bitrate.

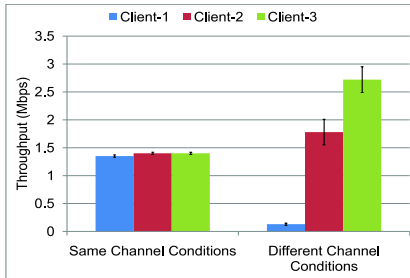


Fig. 7. Impact of different channel conditions client and the BS.

client to reduce its access probability, and thus its available capacity. In addition, a larger fraction of the capacity is used in transmitting transport layer ACKs, which reduces the capacity for data packets. Figure 5 shows the throughput achieved by 5, 10, and 20 clients. When the capacity estimate is inaccurate, client throughputs vary widely. However, when estimation is used, each client achieves a fair bandwidth allocation.

D. Impact of PHY Bitrate

We now analyze the impact of PHY bitrate on the performance of BMCC. Lowering the bitrate increases the transmission time of packets which leads to fewer transmissions in a given interval. This can reduce the accuracy of capacity estimates. Figure 6 shows the throughput of five BMCC clients as a function of the PHY bitrate, where the capacity estimate is set to the bitrate. While inaccurate estimates lead to large throughput variations and degrade fairness, capacity estimation allows BMCC to achieve good performance across all the bitrates allowed by the 802.11b standard. Since lower bitrates are more robust to channel errors, the result suggests that they can result in lower throughput variations than higher bitrates.

E. Impact of Heterogenous Client-BS Channel Conditions

We now evaluate the performance of BMCC, when the channel conditions between each client and the BS are different. We consider three clients that upload data via the BS. The PER of Client-1-BS, Client-2-BS, and Client-3-BS links is 0%, 10%, and 20%, respectively. Observe that clients achieve throughput proportional to the PER they experience (see Figure 7). Clients with poor channel conditions to the BS suffer for two reasons. First, due to more losses, these links experience more retransmissions and larger backoffs at the

MAC layer. Second, clients with good channel conditions get more transmission opportunities, which reduces the air time for transmissions on poor quality links. This result also suggests that the BS can adapt the capacity estimate to provide service differentiation to clients based on application requirements.

VII. CONCLUSION

In this paper, we studied the impact of inaccurate capacity estimates on congestion control protocols over variable capacity links. We showed that inaccurate capacity estimates can significantly degrade performance of protocols that rely on them for feedback computation. We then presented a model for characterizing the capacity in a 802.11 WLAN and studied a method for capacity estimation. Finally, we evaluated the performance of a congestion control protocol using the capacity estimation method. Our results show that the mechanism works well across a range of scenarios. The method is applicable to all shared, multi-access links with a reliable link-layer. In the future, we plan to extend this work to consider multi-hop wireless mesh networks.

VIII. ACKNOWLEDGEMENTS

We would like to thank Fahad Rafique Dogar, Zafar Ayyub Qazi, and the anonymous reviewers for providing useful feedback. This work was supported by the LUMS Faculty Startup Grant, Australian Research Council grant DP1095103, and NSF grants 010536, 010684, and SES-0729456.

REFERENCES

- [1] D. Katabi, M. Handley, and C. Rohrs, "Internet Congestion Control for High Bandwidth-Delay Product Networks," in *ACM SIGCOMM 2002*.
- [2] N. Dukkipati, M. Kobayashi, R. Zhang-Shen, and N. McKeown, "Processor Sharing Flows in the Internet," in *IWQoS 2005*.
- [3] I. A. Qazi, L. L. H. Andrew, and T. Znati, "Congestion Control using Efficient Explicit Feedback," in *IEEE INFOCOM 2009*.
- [4] F. Abrantes and M. Ricardo, "XCP for shared-access multi-rate media," *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 3, pp. 27–38, 2006.
- [5] B. Wyrowski, L. L. H. Andrew, and M. Zukerman, "MaxNet: A Congestion Control Architecture for Scalable Networks," *IEEE Commun. Lett.*, vol. 7, no. 10, pp. 511–513, Oct. 2003.
- [6] I. A. Qazi and T. Znati, "On the design of load factor based congestion control protocols for next-generation networks," *Computer Networks*, vol. 55, no. 1, pp. 45 – 60, 2011.
- [7] J. Pu and M. Hamdi, "Enhancements on router-assisted congestion control for wireless networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 6, pp. 2253–2260, June 2008.
- [8] A. Kashyap, S. Ganguly, and S. R. Das, "A measurement-based approach to modeling link capacity in 802.11-based wireless networks," in *MobiCom 2007*.
- [9] T. Salonidis, G. Sotiropoulos, R. Guerin, and R. Govindan, "Online optimization of 802.11 mesh networks," in *ACM CoNEXT 2009*.
- [10] M. Kazantzidis, M. Gerla, and S.-J. Lee, "Permissible Throughput Network Feedback for Adaptive Multimedia in AODV MANETs," in *ICC 2001*.
- [11] S. H. Shah, K. Chen, and K. Nahrstedt, "Dynamic bandwidth management in single-hop ad hoc wireless networks," *Mob. Netw. Appl.*, vol. 10, no. 1-2, pp. 199–217, 2005.
- [12] L.-J. C. Tony, T. Sun, G. Yang, M. Y. Sanadidi, and M. Gerla, "Adhoc probe: Path capacity probing in wireless ad hoc networks," in *WICON 2005*.
- [13] S. Kunniyur and R. Srikant, "Analysis and Design of an Adaptive Virtual Queue (AVQ) Algorithm for Active Queue Management," in *ACM SIGCOMM*, Aug 2001.
- [14] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE JSAC*, vol. 18, pp. 535–547, 2000.