

Leveraging MAC-layer information for single-hop wireless transport in the Cache and Forward Architecture of the Future Internet

Sumathi Gopal, Sanjoy Paul, Dipankar Raychaudhuri

Wireless Information Network Laboratory (WINLAB), ECE Dept., Rutgers University
Piscataway, New Jersey, USA

Abstract— Cache and Forward (CNF) Architecture is a novel architecture aimed at delivering content efficiently to potentially large number of intermittently connected mobile hosts. It uses a reliable hop-by-hop transport mechanism where in-network storage is leveraged to store the entire file in each node, before forwarding it towards the destination(s). Links between nodes may be wired, or wireless, 802.11 a/b/g, cellular, or satellite, or some other upcoming wired/wireless technology. A key aspect in the CNF architecture is to choose the link-layer protocol with parameters such that it optimizes the performance of the link between two CNF nodes. In this paper we propose Cross-Layer Aware Transport Protocol (CLAP) as a potential candidate for a reliable link-layer protocol between two CNF nodes connected by 802.11 a/b/g link. Leveraging upon MAC layer status information, CLAP quickly adapts its flow control rate to bandwidth fluctuations and maximizes link utilization efficiency. Due to the self-interference problem in wireless links, where ACK packets corresponding to a given flow interfere with the DATA packets of the same flow, CLAP's error control algorithm is based on aggregate negative acknowledgements (NACKs) which reduce the reverse traffic and hence the interference with the data traffic. Since CLAP performs transport layer functionality, we compare it with TCP-SACK - a popular variation of TCP. CLAP significantly outperforms TCP-SACK in both noise-free and noise-prone wireless environments, with throughput gains upward of 150%. For multiple flows, CLAP leverages the fairness characteristics of the underlying MAC layer to achieve fair bandwidth sharing among flows.

Keywords-wireless networking, transport protocols, cross-layer awareness, protocols for the future Internet, Internet Architecture, Cache and Forward

I. INTRODUCTION

This paper addresses reliable file transfer over a single wireless link that gains importance in the Cache and Forward (CNF) architecture of the future Internet. CNF is a novel architecture aimed at delivering content efficiently to potentially large number of hosts some or all of which could be mobile, intermittently connected to the network or connected to the network by multi-hop wireless links as shown in Figure 1. Traditional TCP/IP protocols either do not work or work inefficiently in a network of intermittently connected hosts. CNF architecture overcomes the shortcomings of the TCP/IP architecture by optimizing transport on a hop-by-hop basis. Nodes in the CNF

architecture leverage cheap in-network storage to store an entire file before forwarding the file to the next node, and the idea is to choose a link-layer protocol and its parameters dynamically to optimize the performance of the link between two CNF nodes.

Given the variety of links in the CNF architecture, it may not be efficient to use a single protocol for reliable file transfer. More appropriate would be to dynamically choose a link-layer protocol with parameters best suited for the characteristics of the link. While TCP might be used as a reliable link-layer protocol for certain links, it may not be the most efficient protocol for certain other wireless links. In such links, we might need a different reliable link-layer protocol that avoids the shortcomings of TCP. In fact, TCP primarily addresses the challenges of end-to-end transfer, most dominant of which is network congestion. On the other hand, the challenges in wireless links are significantly different.

Wireless links are characterized by rapidly changing Signal-to-Noise Ratio (SNR) and bandwidth that are caused due to various reasons. SNR fluctuations occur from fading, shadowing and additive noise as revealed by several propagation studies [26]. Thermal noise and other possible noises such as due to co-channel interference cause an additive white Gaussian noise to always be present at the receiver. Bandwidth fluctuations occur due to multiple flows

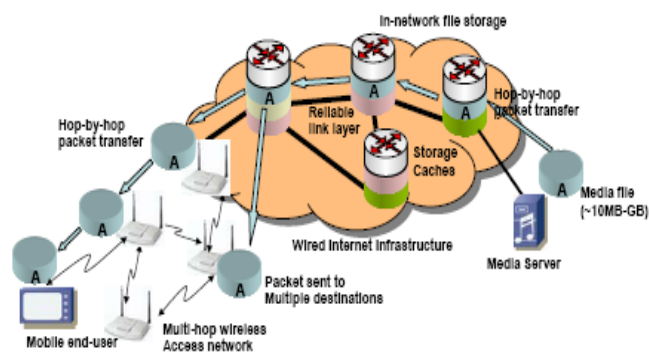


Figure 1: Architecture showing CNF nodes that use in-network storage to store and forward entire files hop-by-hop from the source to the destination(s). CNF nodes exist both in wired as well as in multi-hop wireless networks.

and auto-rate adaptation in wireless cards [10]. Mobility contributes to additional bandwidth and SNR fluctuations. Further due to the inherent broadcast nature, returning feedback packets of the same flow interfere with the data packets - a phenomenon known as self-interference. Thus, link bandwidth fluctuations and high packet error rates are common in wireless links. While the error rates may somewhat be mitigated by error control mechanisms at the link layer, fluctuations in link bandwidth need to be handled by a transport protocol.

We hence propose a cross layer aware transport protocol (CLAP) that specifically addresses these wireless issues. Supplemented by status information from the MAC-layer, CLAP incorporates a rate based flow control that quickly adapts to bandwidth fluctuations. For error control, it uses aggregated negative acknowledgements (NACKs) to reduce feedback traffic, in order to alleviate the self-interference problem. More importantly, the flow control and error control mechanisms are decoupled in CLAP, thus avoiding unnecessary scale down of the flow rate in response to transient packet losses in the wireless link. Results from simulations reveal the tremendous gains of CLAP over TCP and its variants. It substantially outperforms TCP-SACK in both error-free and error-prone wireless scenarios considered, with throughput gains of 240% and 168% respectively. Hence CLAP is an efficient reliable protocol that can be used for transferring an entire file across a single-hop wireless link in the CNF architecture.

In Section II we describe the various wireless link characteristics that motivated the design of CLAP. In Section III we describe the Flow control and error control algorithms used in CLAP. In Section IV, the performance of CLAP in single and multiple flow scenarios is compared with that of TCP-SACK after presenting the simulation setup. In Section V we explain reasons for poor TCP performance over time-varying wireless links and discuss related work in Section VI. Conclusion and Future work are presented in Section VII.

II. CLAP DESIGN OBJECTIVES

CLAP is designed based on the following objectives:

(1) *Address link bandwidth fluctuation in wireless links.* The link bandwidth available to a flow depends upon certain factors. (a) Auto-rate adaptation: The signal modulation used determines the channel rate available between a source-destination pair [26][27]. Most wireless cards have an inbuilt, proprietary auto-rate adaptation scheme. Using this scheme, the link layer selects the modulation most appropriate for the estimated SNR at the destination. Hence in an environment with fluctuating noise conditions, auto-rate adaptation could result in wide fluctuations in link bandwidth [10]. (b) Number of flows contending for the channel: In a contention-based wireless link (such as IEEE 802.11 DCF mode [31]), nodes contend for channel access on a per-packet basis. The channel access time is shared among all contenders.

(2) *Minimize dependence on accurate round-trip time estimation.* Bandwidth fluctuations and high incidence of packet loss hampers accurate round-trip time measurement. Hence the rate adaptation approach in particular, should be designed independent of round-trip time estimates.

(3) *Minimize bidirectional traffic to avoid self-interference.* Interference from returning control traffic such as acknowledgements causes channel access delays and worsens the perceived link quality due to collisions. Reducing the number of returning control packets will conserve the link bandwidth for data packets in the forward direction.

(4) *Decouple flow control from error control.* The data rate supported by a wireless link is determined by the physical modulation used *by the sender*. This is independent of the rate of packet losses that is determined by loss characteristics (noise, collisions etc) affecting *the receiver*. Hence link rate assessment must be performed independent of the number of successfully delivered packets.

A. Incorporating cross-layer information in CLAP

The exact status of a wireless link is known at the physical and link layers. For example, the physical layer is well aware of modulation/channel rate and the average SNR from neighbors. Similarly MAC/Link layer status parameters such as the contention indicator, rate of packet transmission and packet loss rate are of importance at the transport layer. CLAP leverages lower-layer parameters that provide this information to achieve better performance over wireless networks, than traditional transport protocols, such as, TCP.

III. FLOW CONTROL AND ERROR CONTROL IN CLAP

CLAP incorporates 3-way handshake mechanisms for connection establishment and breakdown similar to TCP. The data packet header is the same as that of the NACK packet depicted in Figure 3. In this section, we describe the rate control and error control approaches implemented in CLAP.

A. Flow Control

In designing CLAP, we assume the wireless link to be the sole bottleneck in the end-to-end flow, with no size limitations in either the receiver buffer or the queues in intermediate nodes. *Hence in this paper we use the term "flow control" to refer to adapting the sending rate to network resources.* We also assume that the MAC utilizes a single channel rate for all target destinations.

Rate-based flow control was chosen following design objective #1 (Section II) in order to achieve quick rate adaptation in a dynamical environment. It is known that rate-based flow control achieves faster adaptation, compared to window-based Go-Back-N approaches such as in TCP [6]. However, rate-based flow control requires supplemental information of the initial sending rate and subsequent rates. Rate-control was extensively used in ATM's ABR networks where supplemental rate information was carried in an out-of-band channel. However in earlier IP networks, limited capabilities of nodes could not support such a supplemental

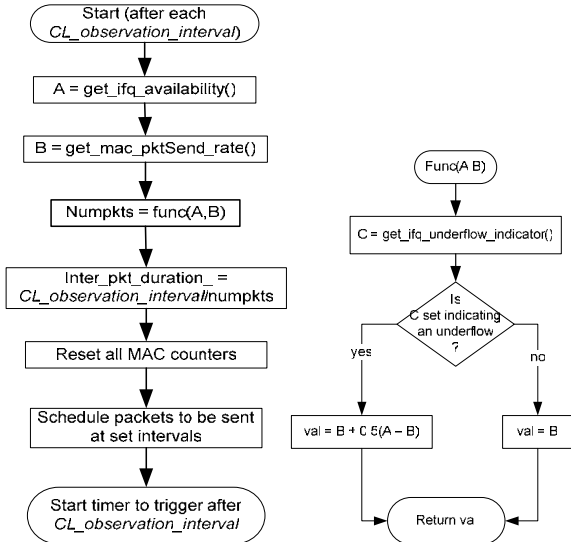


Figure 2: CLAP's flow control algorithm that uses status information from the link layer

plane [6]. Much to the contrary, today's network nodes are well-equipped to sustain a status plane in addition to the data plane motivating this choice in CLAP.

The link layer has sufficient indicators that capture the instantaneous link rate. Specifically the two parameters - MAC packet send rate and Interface queue underflow indicator sufficiently represent the available link rate in the ensuing wireless link. CLAP extracts these values once in every $CL_observation_interval$ and decides the number of packets to be sent in that interval. All packets in this interval are sent at the same packet rate. Packets to be retransmitted are prioritized over new packets. Some simple link layer parameters sufficiently represent the instantaneous link rate available. The MAC packet send rate and a queue underflow indicator can be used in CLAP to gain a handle on the instantaneously available rate. Additional parameters may be required when packet sizes vary.

B. Error Control

A selective reject approach with negative acknowledgements is used for error control in CLAP. The motivation for this approach stems from two factors:

1. Broadcast nature of wireless links over which feedback packets cause self-interference [4].
2. The wireless link layer typically has inbuilt reliability control, such as MAC layer retries in 802.11 or time scheduling in cellular networks. Explicit feedback of the receipt of every single data packet is non-essential at the transport layer.

The receiver sends an *aggregate* NACK to the data sender when an out-of-order packet is received. In this approach, the status of a large number of packets is aggregated in one NACK message. In this paper the term "NACK" refers to "aggregate NACK" unless otherwise specified. Such a feedback mechanism has also been employed in RMTTP for bulk transfer [25]. Timers on the receiver-side ensure at

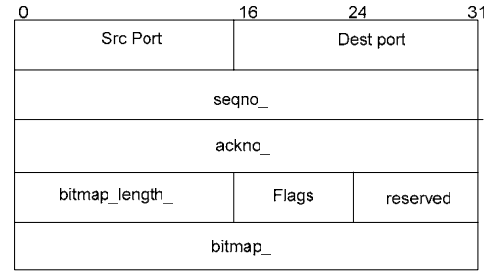


Figure 3 :NACK Packet format

least one NACK is sent in a configurable period. Other optimizations are: a variable bitmap to maximize the feedback per NACK and restricting NACKs to one per round-trip time to minimize duplicates. When the wireless link is good, returning NACKs are primarily due to periodic NACKs, and when the link is error-prone, aggregate NACKs are triggered by perceived packet losses, and the NACKs are limited by the aggregation with the bitmap and the restriction of one NACK per RTT. For multiple flows performance degrades significantly if a NACK restriction is not introduced.

The NACK packet format is depicted in Figure 3. The key fields are those of sequence number ($seqno_$), bitmap ($bitmap_$) and bitmap length ($bitmap_length_$). $seqno_$ indicates the last packet in the correctly received sequence. The $bitmap_$ field indicates the receipt status of $bitmap_length_$ number of packets following $seqno_$. For example, a 0 in the i^{th} -bit of the bitmap indicates correct receipt of packet ($seqno_ + i$), while a 1 indicates the loss of that packet. Results in this paper were obtained by fixing the receiver-side round-trip time estimate to 0.08 seconds. This value was derived from the NS2 trace of successful packets.

IV. PERFORMANCE EVALUATION OF CLAP

In this section, we will describe the simulation setup details and then compare the performance of CLAP with TCP-SACK.

A. Simulation Details

The results in this paper were obtained with simulations in NS2 version 2.1b9a. 802.11b wireless links with a fixed 11 Mbps channel rate in a wireless LAN scenario were considered. The CMU wireless module that implements

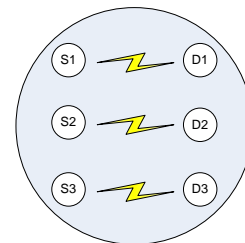


Figure 4: Topology considered in simulations with all nodes within hearing range of each-other

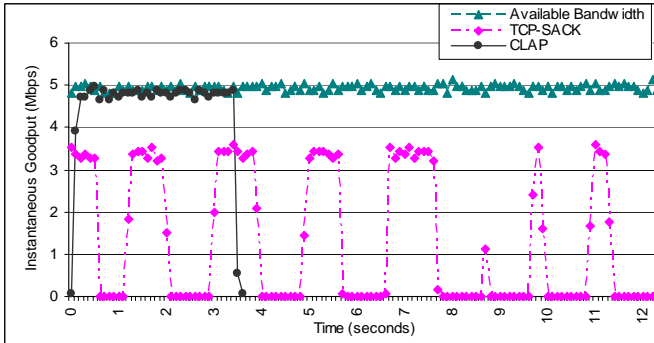


Figure 5: Performance of CLAP and TCP-SACK during a 2MB file transfer over an error-free wireless link. CLAP completes the reliable transfer in 3.6 seconds while TCP-SACK takes over 12 seconds.

802.11 functionality in NS2, was extended to include mechanisms for noise injection required to generate error-prone wireless links affected by AWGN noise.

To simulate time-varying wireless links, a random value for noise power is generated for each incoming packet using a bipolar normal random variable of mean 0 and variance set to 9.3×10^{-8} Watts. The slow-varying wireless link is produced by introducing the random Gaussian noise three times at 1-second intervals. Varying number of simultaneous flows, produce time-variation of link bandwidth

Other simulation parameters:

The wireless nodes were situated in close proximity of each other in order to produce an interference environment. Phy/WirelessPhy object was set with the following parameters - transmit power (P_t) = 0.2818 Watts; bandwidth = 11Mb; dataRate = 11Mb; basicRate = 1Mb; freq = 2.472e9 Hz; CStresh = 5.011872e-12 Watts; RXThresh = 3.652e-10 Watts.

MAC-layer retransmissions were disabled in the course of these experiments, to enable observation of time-varying wireless links. RTS/CTS was also disabled since there were no hidden nodes. A patch that appropriately reset the DeferTimer was applied to correct a persistent NS2 bug that caused an invalid uid_ in the scheduler [34].

TCP's receiver window was set to 10^6 packets, to remove any possible receiver buffer limitations. The interface queue length was set to 100 packets to eliminate any queue overflow situations.

To measure link bandwidth a single CBR flow was used between the same source-destination pair. It comprised of 1000 byte UDP packets at a constant interval of 0.001 seconds. Given the various network stack overheads, the CBR rate of 8Mbps was sufficient to measure the saturating goodput of the wireless link in all the scenarios.

All packet sizes (CLAP, TCP and UDP) were set to 1000 bytes. CLAP header size was set to be 40 bytes, to match that of TCP. Multiple flows were staggered by 0.1 seconds from each other.

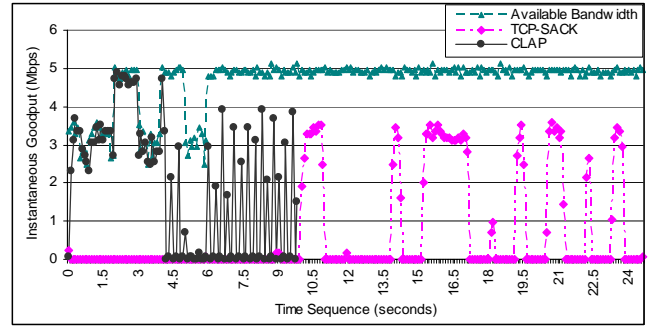


Figure 6: Performance of CLAP and TCP-SACK over a slow-varying wireless link during a 2MB file transfer. CLAP completes the reliable transfer in 9.3 seconds while TCP-SACK takes nearly 25 seconds

B. CLAP vs. TCP-Sack

Although CLAP is used as a link-layer protocol in the CNF architecture, its functionality is that of a reliable transport layer protocol. We hence compare CLAP operation to TCP-SACK which is a popular variant of TCP-Reno that also minimizes reverse traffic with feedback aggregation. We compared performance in error-free as well as certain error-prone scenarios. In both cases, CLAP performed significantly outperformed TCP-SACK, completing the reliable file transfers in a fraction of the time taken by TCP. Some of the results are depicted in Figures 5 and 6. The CLAP flow makes full utilization of the available link bandwidth in the error free environment as shown in Figure 4. The goodput achieved by CLAP in each measurement interval matches that of the CBR goodput. The TCP-SACK goodput on the other hand, significantly falls short of the CBR goodput that measures the bandwidth available in that scenario, despite an error-free wireless environment. TCP-SACK undergoes several timeouts in the course of its file transfer, primarily because of packet losses induced by collisions between its own data and ACK packets.

In the slow-varying scenario considered here, CLAP completed reliable transfer of the 2MB file less than half the time taken by TCP, and achieving a throughput gain of 168%. The gains are much higher as the noise level in the wireless link increases. TCP shuts down operation when the noise is incident, while CLAP makes full use of the link bandwidth. Thus CLAP fully utilizes the wireless link resources despite a noise-prone environment, while TCP significantly underutilizes it.

Multiple Flows:

CLAP's opportunistic rate adaptation adapts to the link bandwidth measured in the previous observation interval. While the status plane *supplies* this information, *no guarantees are made of bandwidth availability* in the network. It might hence appear that CLAP's opportunistic approach could result in an unfair capture of the link by the first flow. This calls for CLAP's performance evaluation with multiple flows.

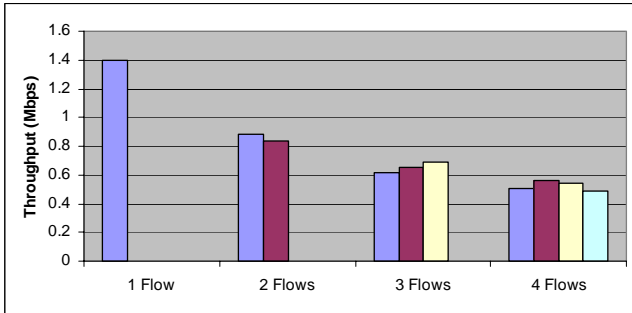


Figure 7: Throughput achieved by each CLAP flow for 1MB file transfer in a noisy fast-varying wireless environment

A plot of throughputs achieved during transfer of 1MB files in a fast-varying link is shown in Figure 7. The plots indicate fair bandwidth sharing among all CLAP flows. This is a direct result of CLAP's leverage of cross layer status information. Channel contention in 802.11 MAC DCF mode [31] is on a per-packet basis and ensures fair bandwidth sharing among flows. Since CLAP matches its own rate to that of the MAC rate, fair bandwidth sharing of the wireless link ensues. Since TCP shuts down operation in noisy environments, one flow may gain throughput over another in a random manner, causing unfairness among flows.

V. REASONS FOR POOR TCP PERFORMANCE

The key reasons for TCP's performance shortcomings over a wireless link lie in the core design of TCP:

Rate control and error control are tightly coupled. TCP implements the AIMD (Additive Increase, Multiplicative Decrease) algorithm that combines flow rate control and error control operations. It halves its sending rate (by halving the congestion window size) in response to a packet loss, irrespective of the reason for the packet loss. Subsequently the flow rate is improved cautiously in a linear manner with respect to the round-trip time. This design approach of TCP stems from the characteristics of wired networks, where router queues are assumed to be the sole bottlenecks. Over wireless links however, there is a tremendous performance loss due to this behavior.

TCP's flow control works on a "macroscopic" end-to-end view and misses the "microscopic" view of link changes. It thereby fails to leverage good condition of the links in the slow-varying link case as shown in Figure 6.

TCP depends on the timely arrival of positive acknowledgements. The high-incidence of packet losses in error-prone wireless links, arrests growth of the congestion window and hence the packet sending rate. It also hampers the accurate estimation of round-trip time that is crucial to set TCP timers. In Figure 6, this is the reason why TCP stops sending data when the link is noisy.

TCP's bidirectional traffic causes self-interference. Self-interference is a phenomenon unique to CSMA/CA-based wireless links, where packets of the same flow traversing opposite directions compete for channel access [4]. Wasted transmissions (due to collision or simultaneous-send) result if more than one node transmits at the same time.

The operation of TCP in throughput spikes in an error-free link (In Figure 6, this is the region after the noise has ended), is an artifact of TCP's self-interference that is outside the scope of this paper.

VI. RELATED WORK

There are several papers written that address transport problems in wireless networks. They primarily address a specific problem in cellular networks and MANETs. Transport problems over time-varying wireless links are less understood.

In AIRMAIL [33], the wireless link-layer is made reliable to make it transparent to TCP, but this requires modification of the link layer. Ideas similar to this were used in Radio Link Protocol (RLP).

Similar to CLAP, the Adhoc Transport Protocol (ATP) [3] also uses a rate-based approach for flow control. However, the rate adaptation still depends upon the frequency of returning (positive) acknowledgements like in TCP. Hence in a noise-prone wireless link, repeated loss of returning acknowledgements would slow down the packet sending rate, despite good bandwidth availability.

A rate-based flow control scheme for multihop wireless networks is explored by Jayasuriya et. al. in [28]. However similar to ATP this scheme also depends on regular feedback from the destination and does not operate well when the link is error-prone.

There are various variations of TCP proposed for wireless networks, that offer *multi-layer* solutions [14][16][17][23]. Their approach is to hide wireless packet losses from the link layer, and thus prevent timeouts in TCP. This function is performed by a proxy agent in an intermediate node, that does not apply to the CNF architecture problem addressed in this paper.

VII. CONCLUSION AND FUTURE WORK

We have demonstrated the unquestionable gains of CLAP when operated over a single wireless link. CLAP also represents one of the early attempts of a cross-layer approach to improve reliable data transport over wireless networks. The protocol is particularly promising because of its ability to address various wireless network challenges such as self-interference [4][5], fast-varying link SNR, auto-rate adaptation and MAC contention due to multiple flows. Despite its light-weight design it opportunistically adapts to large fluctuations in link bandwidth despite a high incidence of packet errors.

Since network nodes in the CNF architecture are designed to be highly configurable, CLAP and the corresponding status-providing elements in the network stack, can easily be installed in them. CLAP's modularity in the data plane enables it to operate over any underlying network technologies, thus making it a generic transport protocol.

Work is ongoing to extend the provision of status information to multiple nodes, so as to extend CLAP's operation to the generic multihop scenario. We are also

implementing CLAP in the linux kernel to be part of the ORBIT 400-node testbed [7].

REFERENCES

- [1] K. Chen, K. Nahrstedt, N. Vaidya, "The Utility of Explicit Rate based flow control in Mobile Ad Hoc Networks", WCNC 2004
- [2] Z. Fu, P. Zerfos, H. Luo, S. Lu, L. Zhang, and M. Gerla, "The impact of multihop wireless channel on tcp throughput and loss," in *Proc. IEEE Infocom 2003*, San Francisco, California, USA, Apr. 2003.
- [3] Sundaresan, V. Anantharaman, H.-Y. Hsieh, and R. Sivakumar. ATP: A reliable transport protocol for ad-hoc networks. In *Proceedings of 4th ACM MobiHoc*, pp. 64–75, 2003.
- [4] S. Gopal, S. Paul, D. Raychaudhuri, "Investigation of the TCP Simultaneous Send problem in 802.11 Wireless Local Area Networks", IEEE ICC 2005, May 16-20, Seoul, South Korea.
- [5] S. Gopal, D. Raychaudhuri, "Experimental Evaluation of the TCP Simultaneous Sent problem in 802.11 Wireless Local Area Networks", ACM SIGCOMM Workshop on Experimental Approaches to Wireless Network Design and Analysis (EWIND-05), August 22nd 2005, Philadelphia.
- [6] D.D.Clark, M.L. Lambert, L. Zhang, "NETBLT: a High Throughput Transport Protocol", ACM SIGCOMM 1987
- [7] ORBIT: Open Access Research Testbed for Next-Generation Wireless Networks www.orbit-lab.org
- [8] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu and M. Singh "Overview of the ORBIT Radio Grid Testbed for Evaluation of Next-Generation Wireless Network Protocols", to appear in the Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC 2005)
- [9] Vikas Kawadia and P. R. Kumar, "A Cautionary Perspective on Cross Layer Design." IEEE Wireless Communication Magazine. pp. 3-11, vol. 12, no. 1, February 2005
- [10] Z. Wu, S. Ganu, I. Seskar, D. Raychaudhuri. " Experimental Investigation of PHY Layer Rate Control and Frequency Selection in 802.11-based Ad-Hoc Networks", ACM SIGCOMM Workshop on Experimental Approaches to Wireless Network Design and Analysis (EWIND-05), August 22nd 2005, Philadelphia
- [11] M C Chan and Ramchandran Ramjee, "TCP/IP performance over 3G wireless link with rate and delay variation", ACM MOBICOM 2002
- [12] K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash. "A feedback-based scheme for improving TCP performance in ad hoc wireless networks", *IEEE Personal Communications Magazine*, 8(1):34–39, Feb. 2001
- [13] T.Goff, J. Moronski, D.S. Phatak, V. Gupta, Freeze-TCP: A true end-to-end TCP enhancement mechanism for mobile environments, INFOCOM 2000
- [14] A Bakre and B. Badrinath, "I-TCP: Indirect TCP for Mobile Hosts," in Proc. Of 15th International Conference on Distributed Computing Systems (ICDCS), Vancouver, Canada May 1995
- [15] Xin Yu, "Improving TCP Performance over Mobile Ad Hoc Networks by Exploiting Cross-Layer Information Awareness" in Proceedings of MOBICOM 2004
- [16] G. Holland and N.H. Vaidya, "Impact of Routing and Link Layers on TCP Performance in Mobile Ad-hoc Networks", Proc. IEEE WCNC, Sept 99
- [17] V. Anantharaman and R. Sivakumar, "A microscopic analysis of TCP performance over wireless ad-hoc networks", Presented in 2nd ACM SIGMETRICS (Poster Paper), 2002.
- [18] M. Chinta, A. Helal, "ILC-TCP: An Interlayer Collaboration Protocol for TCP Performance Improvement in Mobile and Wireless Environments", WCNC 03
- [19] P. Sudame and B.R. Badrinath, "On providing support for protocol adaptation in mobile wireless networks", Mobile Networks and Applications 2001, Vol.6, No. 1, pp. 43-55.
- [20] K.Chen, S. H. Shan and K. Nahrstedt, "Cross-Layer design for data accessibility in mobile ad hoc networks" Wireless Personal Communications '02
- [21] L. Xu, K Harfoush, I Rhee "Binary Increase Congestion Control (BIC) for Fast Long-Distance Networks", In Proceedings of IEEE INFOCOM, 2004
- [22] M. Conti, G. Maselli, G. Turi, S. Giordano, "Cross-layering in mobile ad hoc network design", Computer Volume 37, Issue 2, Feb 2004 Page(s): 48 - 51
- [23] H. Balakrishnan, S. Seshan, E. Amir and R. Katz, "Improving TCP/IP performance over wireless networks", Proceedings of ACM MOBICOM, Nov 1995
- [24] J. Liu and S. Singh, "ATCP: TCP for Mobile Ad Hoc Networks", IEEE Journal on Selected Areas in Communications, 2001
- [25] J.C. Lin, S. Paul "RMTP: A Reliable Multicast Transport Protocol", IEEE INFOCOM, 1996
- [26] Theodore S. Rappaport, "Wireless Communication, Principles and Practice", Second Edition, Prentice Hall Inc., 2002
- [27] John Proakis, "Digital Communications", Fourth Edition, McGraw Hill Companies, 2001.
- [28] Y. Liaw, A. Dadej, A. Jayasuriya, "Rate-based End-to-end Flow Control for Multi-hop Wireless Networks", in Proc. 7th IFIP International Conference on Mobile and Wireless Communication Networks (MWCN 2005), Sept 2005
- [29] G. Gaertner, E. ONuallain, A. Butterly, K. Singh, V. Cahill. "802.11 Link Quality and its Prediction - An Experimental Study", Personal Wireless Communications (PWC) 2004
- [30] Sanjit Kaul, Marco Gruteser, and Ivan Seskar, "Creating Wireless Multi-hop Topologies on Space-Constrained Indoor Testbeds Through Noise Injection" , Proceedings of IEEE Tridentcom 2006, Barcelona, Spain, March 1-3, 2006
- [31] IEEE 802.11, 1999 Edition (ISO/IEC 8802-11: 1999) Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications
- [32] The CMU Monarch Project's Wireless and Mobility extensions to NS: <http://www.monarch.cs.cmu.edu/>
- [33] E. Ayanoglu, S. Paul, T. F. LaPorta, K. K. Sabnani and R. D. Gitlin, "AIRMAIL: A link-layer protocol for wireless networks, Wireless Networks", Volume 1, Number 1, March 1995
- [34] NS2 Bug Fixes - Zhibin Wu, WINLAB, Rutgers University - "http://www.winlab.rutgers.edu/~zhibinwu/html/network_simulator_2.html"