

Cross-layer analysis of TCP performance in a 4G system

Stefan Alfredsson[†], Anna Brunstrom[†] and Mikael Sternad[‡]

[†]Department of Computer Science, Karlstad University, SE-651 88 Karlstad, Sweden

[‡]Signals and Systems, Uppsala University, SE-751 20 Uppsala, Sweden

Email: {Stefan.Alfredsson,Anna.Brunstrom}@kau.se, Mikael.Sternad@signal.uu.se

Abstract—This paper presents results from an experimental study of TCP in a wireless 4G evaluation system. Test-bed results on transport layer performance are presented and analyzed in relation to several link layer aspects. The aspects investigated are the impact of channel prediction errors, channel scheduling, delay, and adaptive modulation switch level, on TCP performance. The paper contributes a cross-layer analysis of the interaction between symbol modulation levels, different scheduling strategies, channel prediction errors and the resulting frame retransmissions effect on TCP. The paper also shows that highly persistent ARQ with fast link retransmissions do not interact negatively with the TCP retransmission timer even for short round trip delays.

1. INTRODUCTION

The demand for higher capacity and wider coverage of mobile wireless network access is increasing. As third generation mobile systems are commercialized, the research focus has shifted towards systems for higher data rates, “4G”. One promising technology for 4G is orthogonal frequency division multiplexing (OFDM), which uses multiple carrier frequencies dedicated to a single data source.

One 4G system proposal based on OFDM has been developed within the “Wireless IP” project at Uppsala University, in cooperation with Chalmers University of Technology and Karlstad University [1]. The main focus is to cover wide areas to service vehicular users, in excess of speeds of 100 km/h with a 30-fold bandwidth increase compared to UMTS/3G. To realize this goal, adaptive OFDM is used in combination with channel prediction. Transmissions can then be scheduled to maximize the total satisfaction of the users, depending on their current channel quality. This is combined with increased cross layer interaction, link level ARQ, and other mechanisms.

The Wireless IP evaluation system is conceived as an all-IP system intended for Internet traffic. It should be designed to provide a good service to the network and transport layers and it is thus important to consider the system level implications of lower layer design decisions. In order to allow performance measurements on the interaction between the physical/link layer design and upper layers the Wireless IP emulator (WIPEMU) was developed. Details on the emulator are available in [2].

In this paper, we present the results and analysis from a number of experiments on this platform. A central concept of the 4G system is the channel prediction used to schedule

transmissions and choose the modulation scheme. The investigation is focused on the impact of channel prediction errors on the link layer performance and the transport layer performance. We also present results on the symbol modulation level utilization, and how different wireline network delays affect the TCP performance.

2. RELATED WORK

In previous research on TCP over wireless, many studies and suggested improvements focus on the issue of non-congestion related packet loss: Packets lost due to link errors are assumed to be lost due to congestion by TCP and the sender initiates congestion avoidance unnecessarily. Examples of these suggested improvements include TCP-DCR [3], Explicit Loss Notification [4], Syndrome [5], Explicit Bad State Notification [6], and Partial Acknowledgements [7]. Other approaches hide the unreliable link by performing local TCP retransmissions, such as Snoop [8] and WTCP [9]. More background on these issues can be found in [10], [11].

The mentioned approaches assume that link errors translate into packet loss. For a reliable link (obtained e.g. by persistent link retransmissions), these events would not occur. However, using link retransmissions leads to delay variations, which, depending on the RTO timing, may either be interpreted as TCP packet loss or cause RTO inflation. To this end, approaches such as Eifel [12] and TCP-Westwood [13] have been proposed to handle delay spikes (among other improvements).

The Wireless IP evaluation system can use a varying degree of link reliability, by configuring the maximum number of allowed link retransmissions. In this paper highly persistent retransmissions are configured to provide a reliable link layer¹. However, the link retransmissions are very fast (2 ms, which is three times the link frame transmission time), which should contribute to low delay variation. This paper investigates the performance of regular TCP over such a link.

3. EXPERIMENTAL SETUP

3.1 Scenario

The use case considered in the experiments consists of a mobile user downloading content from a server using TCP.

¹As presented in [14], a reliable link layer provides the best performance considering best-effort traffic.

The user has a wireless connection to a base station, which in turn is connected to the rest of the Internet, which also connects the server. This scenario is shown in Figure 1. The scenario is emulated with three computers, one sender, one gateway and one receiver. The gateway runs the WIPEMU software and all nodes collect packet traces for later analysis.

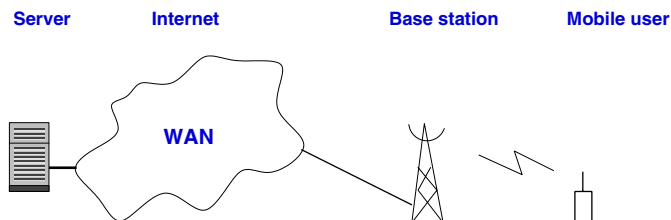


Fig. 1. Logical experiment setup.

3.2 The Wireless IP evaluation system

The Wireless IP evaluation system assumes the use of frequency division duplex (FDD), and the use of a base station infrastructure with sectorized antennas. User terminals are in this paper assumed to have only one antenna.

The available downlink bandwidth within a sector (cell) is assumed to be slotted in time. Each slot of duration T is furthermore partitioned into *time-frequency bins* of bandwidth Δf_b . We here assume $T = 0.667$ ms and $\Delta f_b = 200$ kHz, which is appropriate for stationary and vehicular users in urban or suburban environments [15]. Each bin of 0.667 ms \times 200 kHz carries 120 symbols. Of these 120 symbols, 12 are allocated for training and downlink control, leaving 108 payload symbols [16].

During slot j , each terminal predicts the signal to interference and noise ratio (SINR) for all bins, for a prediction horizon that equals the time delay of the transmission control loop. During slot $j + 1$, all terminals that may have data to receive then signal their predicted signal quality on an uplink control channel. A scheduler that is located on the network (base station) side of the link then allocates the time-frequency bins exclusively to different users during slot $j + 2$. The payload transmission is performed during slot $j + 3$ and control information that specifies the allocation is transmitted during the same slot. The downlink transmission delay over the air interface, with an empty transmit buffer, is thus three slots or 2 ms. This is a conservative number, as an efficient scheduler may operate before slot $j + 2$ in which payload would then be transmitted in two slots or 1.33 ms.

For the payload symbols, an adaptive modulation system is utilized that uses 8 uncoded modulation formats: BPSK, 4-QAM, cross-8-QAM, 16-QAM, cross-32-QAM, 64 QAM, cross-128 QAM and 256 QAM. All payload symbols within the bin use the same modulation format, which is adjusted to the SINR that is predicted to occur during the utilized bin. We here consider prediction of the channel gain within each

bin, while the interference and noise is modeled as additive Gaussian white noise, with known variance.

OFDM channel predictors can be designed to work either in the time domain [17] or in the frequency domain [16]. As data, they use measurements at the location of known pilot symbols and as prior information, they use the statistics of the channel variations over time and space, given by the Doppler spectrum and the power delay profile or the channel correlation function in the frequency domain. Channel power gain estimates that are based on Kalman prediction of the complex-valued channel provide the best performance, see [18] for a survey. To be useful, channel predictors must provide not only point estimates of the channel power gain, but also a measure of the prediction accuracy. The attainable prediction accuracy depends on the SINR, the downlink pilot symbol density, the vehicle velocity, the carrier frequency, the fading environment and the required prediction horizon. A convenient accuracy measure, used here, is the Normalized Mean Squared Error (NMSE) of the complex-valued frequency domain channels. For reasonable vehicle speeds and SINRs, prediction NMSE values are in the range 0-0.1 for the considered system when it works at a 2 GHz carrier frequency.

The SINR measurements and their accuracy estimates are used to determine the modulation format. For a known prediction NMSE, the SINR ranges in which the different modulation formats are used can be adjusted to attain a performance criterion. This investigation uses the criterion that the number of bits transmitted in correctly received payload bits within a bin is maximized. This maximizes the throughput when using a retransmission scheme with infinite persistence. This adjustment is presented in [19]. For a given NMSE, it is obtained by balancing the number of bits received, which increases for higher modulation formats, against the risk of obtaining bit errors within the frame. An increasing prediction NMSE will tend to reduce the attainable throughput.

Channel quality predictions can be used also to affect the scheduling, i.e. the allocation of bins to users. We will here use a class of schedulers that provides a guaranteed resource share, by allocating precisely one bin to the considered user within each time slot. Two variants of this scheduler are compared:

- 1) "Static allocation", which always allocates the same bin to the user.
- 2) "Scheduled allocation", which allocates the bin with the largest predicted SINR among all predicted bins to the user.

3.3 WIPEMU emulator implementation

The downlink of the Wireless IP evaluation system is emulated by WIPEMU in the following way: IP packets are captured at the gateway and placed in a queue awaiting transmission. A "link frame" of duration 0.667 ms is transmitted with rate 1500 Hz. First a predicted channel is examined to

determine the modulation level. The corresponding amount of bits are then put in the link frame, and “transmitted”. During transmission, a symbol error probability is calculated depending on the modulation level and the non-predicted (“real”) channel. A frame with at least one symbol error is then marked as erroneous, and is retransmitted after an adjustable delay. We below use a retransmission delay of 2 ms, or 3 frames. When the frame is retransmitted, it may be that the channel conditions have changed, and a new modulation is needed. If a lower modulation must be used, the frame is split into two. If a higher modulation can be used, we instead use the original modulation. This provides better resilience to errors, but wastes some efficiency. Better throughput could be obtained by allocating more data in the frame, but this would require more overhead information within the frames to be able to separate data blocks as they would no longer be consecutive.

After all frames have been transmitted successfully, the IP packet is released from the gateway to the destination. More details on the implementation are available in [2].

3.4 Parameterization

The system parameters used in the experimental evaluation were set as follows: Both the sender and the receiver run Linux 2.6.8 and we use the default available TCP version. The standard TCP settings are used except that the timestamp option is not used. The retentive TCP caching available in Linux is turned off to avoid creating any dependencies between the experimental runs. A standard MTU of 1500 bytes is used.

For the wireless channel properties, the receiver and transmitter are assumed to have single antennas. Data for 25×200 kHz channels was obtained from a Jakes model with 12 taps set. It was sampled at 5 Mhz and used a power delay profile according to the typical urban fading (-5.7 , -7.6 , $-inf$, -4.4428 , -13.4 , $-inf$, -13.5793 , -14.2371 , -14.4794 , -16.9543 , -20.0164 , -24.3) at a speed of 75 km/h. To account for shadow fading, an AR(1) process provided shadow samples at an interval of 2 meter. The standard deviation, σ , was set to 4 dB and the pole, a , at 0.74. These channels were then processed to account for a range of prediction errors, thus resulting in a “real” version and several “predicted” versions of each channel. On these channels the two channel allocation schemes (static and scheduled), outlined earlier, were applied.

For the wired part of the network, there is a set round-trip delay of 80 ms. In the experiments with varying delays this delay varies between 2 and 1200 ms. In addition, delays are introduced by packet queuing in the gateway (limited to 50 packets), and the transmission delay in the wireless link layer. These last two delays will vary, because of queue build up, and because the delay in the wireless link layer depends on both retransmissions and varying modulation levels. Delays due to hand-over have not been introduced. The uplink is

emulated to have a 20 kbit/s uplink and 2 ms delay (via the FreeBSD dummynet functionality [20]).

Table I contains a compilation of the relevant parameters for the wired network, the wireless network and the protocol settings.

Wired network	
Wired network delay	80 ms (RTT) when fixed
Network queue size	50 packets
Wireless Downlink	
Frame transmission delay	0.667 ms
Channel model	12-tap Jakes typical urban fading model @ 75 km/h, 16 dB SINR + AR(1) shadow fading with variance 4 dB and pole at 0.74
Modulation	Adaptive modulation (1-8 bits/sym) with switching adjusted for perfect prediction or a prediction error of NMSE 0.1 [19]
Frame size	108 symbols
Coding	Uncoded M-QAM used
Scheduling	Either static or best channel allocation
Link ARQ policy	Infinite persistence, no soft combining
Wireless uplink	
Channel model	imposed bandwidth limit and delay
Packet loss	0% (lowest modulation level assumed)
Capacity	20 kbit/s
Delay	2 ms
Transport layer parameters	
Protocol	TCP (Linux 2.6.8)
Transferred data	3 Mb bulk data
TCP settings	Standard, except for disabled timestamp
MTU	Standard, 1500 bytes
Retentive TCP caching	Cleared before new connections

TABLE I
EXPERIMENT PARAMETERS

4. RESULTS

Studying cross-layer interaction is a complex task, as there are many parameters involved. The study focuses on the parameters most relevant to the considered transmission performance. For TCP, the relevant performance metric is the achieved throughput. The throughput is affected by the capacity of the link, retransmissions on both link- and transport layer, and delay. Therefore, to understand the behavior of TCP performance, we systematically vary parameters affecting the capacity (in form of channel prediction error, scheduling strategies and delay), while studying the cross-layer effects.

4.1 TCP performance

For the transport layer, the throughput is investigated with regard to varying channel prediction errors, channel scheduling, and with regard to the delay in the fixed network.

Throughput vs channel prediction: Figure 2 shows the TCP throughput as a function of a varying degree of prediction errors. One 200 kHz bin is allocated per time slot, at 16 dB average SNR. The experiments evaluated both static and scheduled channel allocation, together with adaptive modulation levels optimized for two different target prediction

error NMSEs. First we use target NMSE 0, implying zero prediction error (i.e. perfect prediction). Then we use target NMSE 0.1 which is a realistic prediction performance when predicting 0.1-0.3 carrier wavelengths ahead [19]. This prediction distance would be required at vehicular velocities in the Wireless IP evaluation system at 2 GHz carrier frequency for prediction horizons of 2 slots.

The graph in Fig. 2 shows three interesting aspects. First we note the difference between scheduled and static channel allocation. Using scheduled channel allocation always results in better performance than the static one. This is not surprising, since the scheduler has chosen the frames of highest SINR for transmission.

We further note that the performance is best when we have a perfect prediction (channel NMSE 0 to the far left in the graph). As the prediction error is increased, the performance drops. The reason for this is that as we get more errors in the prediction, the more symbol errors occur. This leads to link frame retransmissions, which increase the delay and reduce capacity. This increase of link retransmissions is further explored in the next subsection.

Third, we note that the modulation levels optimized for NMSE 0 performs worse than those optimized for NMSE 0.1, except when the channel is perfectly predicted. This hold even for small prediction errors. This is explained by the fact that levels for NMSE 0 are more “aggressive” since they assume a tighter variation of the prediction error bounds. When the error is larger, more link frames become erroneous, leading to more frame retransmissions.

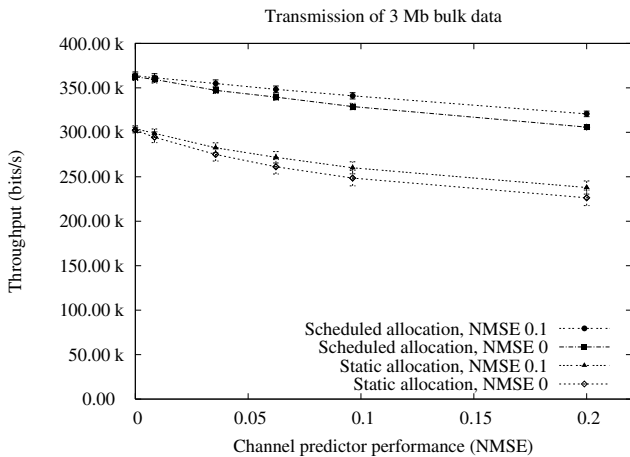


Fig. 2. TCP throughput as a function of the channel predictor performance, for different scheduling strategies.

Throughput vs network delay: The impact of the wired network delay on the TCP performance is shown in Figure 3 for a prediction error of NMSE 0.1. Here the vertical axis shows the throughput as a function of the RTT in the wired network. Like above, the same differences between static and scheduled allocation, and between different adap-

tive modulation targets, exist. An interesting observation is that the throughput is not severely affected by RTT’s up to several hundred milliseconds. This indicates that the link retransmissions are fast enough to not interfere with TCP, even when the RTT in the fixed network is low. As the delay increases, the throughput decreases, due to the bandwidth-delay product becoming larger than the receiver window.

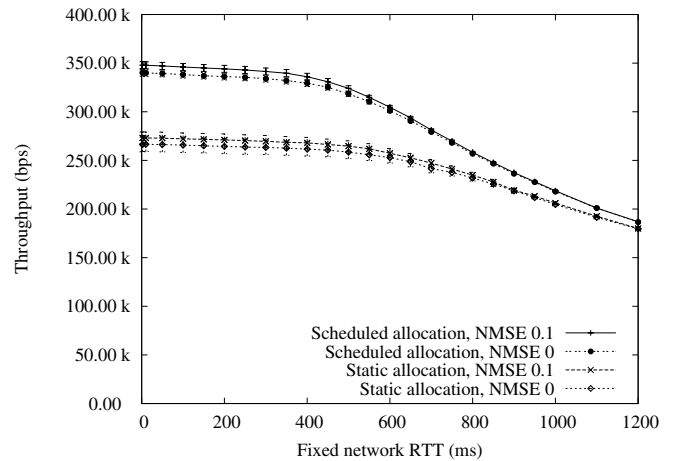


Fig. 3. TCP throughput as a function of the fixed network delay.

4.2 Link performance

For the link layer, the amount of retransmissions and modulation level utilization is investigated, and correlated to the transport layer performance.

Frame retransmissions: Figure 4 shows the average link frame retransmissions (vertical axis) with a varying degree of prediction errors (horizontal axis). The data shown is from the same experiments described in the previous subsection on TCP performance. First we note that we always have retransmissions occurring. This is normal, as some degree of retransmissions are needed to get optimal capacity.

Second, we see that as the prediction error increases, more frame retransmissions are needed. As more link retransmissions are used, the throughput naturally decreases on the transport layer, as seen in Fig 2. Third, there is a big difference in the number of retransmissions between scheduled and static channel allocation. The primary reason for this is that the static channel allocation scheme sometimes uses bins with a SINR below the limit for the lowest modulation level. This leads to an increased amount of retransmissions, compared to the scheduled allocation. Fourth, there is also a difference between using different switch-levels in the adaptive modulation schemes, although not as significant as for channel allocation. This is because of the earlier mentioned aggressiveness when a tighter bound on the prediction error is expected, but not achieved.

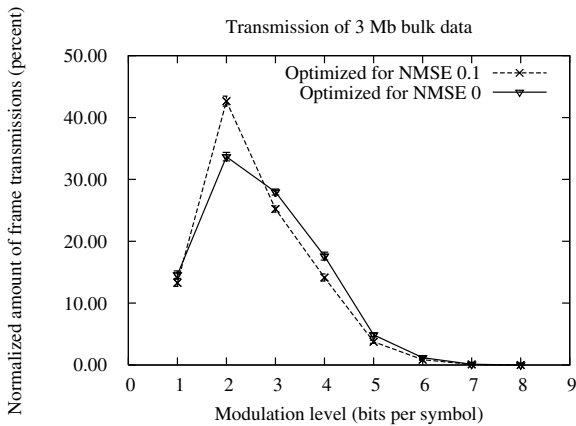


Fig. 5. Modulation level utilization for scheduled channel allocation with channel prediction NMSE 0.1

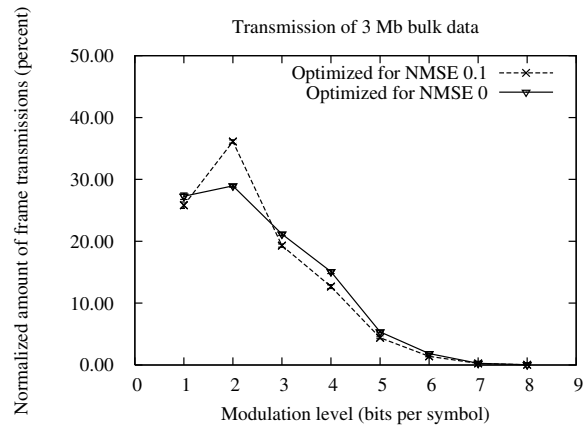


Fig. 6. Modulation level utilization for static channel allocation with channel prediction NMSE 0.1

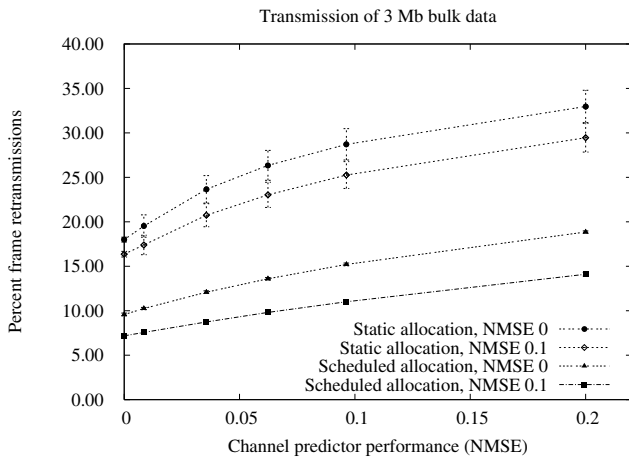


Fig. 4. Average number of link frame retransmissions, as a function of the channel predictor performance.

Modulation utilization: To show the root cause for the increasing link frame retransmissions we investigate how the utilized modulation levels vary with the switch level optimizations, for the scheduled and static allocation schemes. Figures 5 and 6 show the normalized amount of link frames that has used a specific modulation level (from BPSK to 256-QAM).

For all scenarios, we see a spike at 2 bits per symbol (4-QAM). This gives an indication of the mean channel SINR, and if there was a better mean SINR, the spike would shift to the right.

Examining the figures, we see the previously mentioned aggressiveness for switch levels optimized for a target prediction errors of NMSE 0 compared to a target prediction error of NMSE 0.1. This is indicated by the increased use of higher modulation (3 bits/symbol and up) levels by NMSE 0. The more conservative NMSE 0.1 levels instead transmit

more frames with 2 bits/symbol. This then results in fewer retransmissions, as is seen in Fig 4 by comparing the NMSE 0 and NMSE 0.1 curves.

Comparing the scheduled allocation to the static allocation, we also see the cause of the capacity difference between scheduled and static channel allocation. For scheduled allocation in Fig 5, about 13% of the frames use 1 bit/symbol modulation. For static allocation in Fig 6, about 27% of the frames now use this lower modulation, leading to lower capacity and lower throughput.

We now know how the transport layer performance depends on the link layer performance, in terms of retransmissions, and their origin in utilization of varying modulation levels.

5. CONCLUSIONS

This paper has presented results on TCP performance in a 4G evaluation system, with a cross layer perspective. Experiments have been performed with a varying amount of channel prediction errors, scheduled and static channel allocation, and two different optimizations for the adaptive modulation switching levels. The results show how the modulation on the link layer contribute to link frame retransmissions, which give rise to varying TCP performance. The fast link retransmissions do not interact negatively with the TCP retransmission timer, even for short delays in the fixed network. Further, the results show the positive impact of scheduled channel allocation and optimization of the adaptive modulation switching levels.

References

- [1] M. Sternad, T. Ottoson, A. Ahlén, and A. Svensson, "Attaining both coverage and high spectral efficiency with adaptive OFDM downlinks," in *Proceedings of IEEE Vehicular Technology Conference VTC2003-Fall*, Orlando, FLA, Oct 2003.
- [2] S. Alfredsson, A. Brunstrom, and M. Sternad, "Transport protocol performance over 4G links: Emulation methodology and results," in *Proceedings of ISWCS'06*, September 2006.

- [3] S. Bhandarkar, N. E. Sadry, A. N. Reddy, and N. H. Vaidya, "TCP-DCR: A novel protocol for tolerating wireless channel errors," in *IEEE Transactions on Mobile Computing*, vol. 04, no. 5, pp. 517-529, Sept/Oct, 2005.
- [4] H. Balakrishnan and R. Katz, "Explicit loss notification and wireless web performance," in *Proceedings Globecom Internet Mini-Conference, Sydney, Australia*, November 1998.
- [5] W.-P. Chen, Y.-C. Hsiao, J. C. Hou, Y. Ge, and M. P. Fitz, "Syndrome: a light-weight approach to improving TCP performance in mobile wireless networks," *Wireless Communications and Mobile Computing*, no. 2, pp. 37-57, 2002.
- [6] B. Bakshi, P. Krishna, N. H. Vaidya, and D. K. Pradhan, "Improving performance of TCP over wireless networks," *Proceedings 17th Int. Conf. Distributed Computing Systems, Baltimore, USA*, May 1997.
- [7] S. Biaz, M. Mehta, S. West, and N. H. Vaidya, "TCP over wireless networks using multiple acknowledgements," *Technical Report 97-001, Computer Science, Texas A&M University*, January 1997.
- [8] H. Balakrishnan, S. Seshan, and R. H. Katz, "Improving reliable transport and handoff performance in cellular wireless networks," *ACM Wireless Networks*, vol. 1, no. 4, December 1995.
- [9] K. Ratnam and I. Matta, "WTCP: An efficient transmission control protocol for networks with wireless links," in *Proceedings of the third IEEE Symposium on Computers and Communications (ISCC '98), Athens, Greece*, June 1998.
- [10] A. Wennström, S. Alfredsson, and A. Brunstrom, "TCP over wireless networks," *Karlstad University, Karlstad University Studies* 2004:21, 2004.
- [11] K. Xu, Y. Tian, and N. Ansari, "Improving TCP performance in integrated wireless communications networks," *Computer Networks*, vol. 47, no. 2, pp. 219-237, Feb 2005.
- [12] R. Ludwig and M. Meyer, "RFC 3522: The Eifel detection algorithm for TCP," 2003.
- [13] C. Casetti, M. Gerla, S. Mascolo, M. Sansadidi, and R. Wang, "TCP Westwood: End-to-end congestion control for wired/wireless networks," *Wireless Networks*, vol. 8, no. 5, pp. 467-479, September 2002.
- [14] S. Alfredsson, A. Brunstrom, and M. Sternad, "Emulation and validation of a 4G system proposal," in *Proceedings of Radiovetenskap och Kommunikation, Linköping, Sweden*, June 2005.
- [15] W. Wang, T. Ottosson, M. Sternad, A. Ahlén, and A. Svensson, "Impact of multiuser diversity and channel variability on adaptive OFDM," in *Proceedings of IEEE Vehicular Technology Conference VTC2003-Fall, Orlando, FLA, Orlando, FLA*, Oct 2003.
- [16] M. Sternad and D. Aronsson, "Channel estimation and prediction for adaptive OFDM downlinks," in *IEEE VTC 2003-Fall*. Orlando, Fla, Oct. 2003, 2003.
- [17] T. Ekman, M. Sternad, and A. Ahlén, "Unbiased power prediction on broadband channels," in *IEEE Vehicular Technology Conference VTC2002-Fall, Vancouver, Canada*, Sep 2002.
- [18] D. Aronsson, "Channel estimation and prediction from a bayesian perspective," *Licentiate thesis, Signals and Systems, Uppsala University, June 2007*, 2007.
- [19] M. Sternad and S. Falahati, "Maximizing throughput with adaptive M-QAM based on imperfect channel predictions," in *Proceedings of IEEE PIMRC, Barcelona*, Sep 2004.
- [20] L. Rizzo, "Dummysnet: A simple approach to the evaluation of network protocols," *ACM Computer Communication Review*, vol. 27, no. 1, pp. 31-41, January 1997.