Cross-Layer Wireless Bit Rate Adaptation

Mythili Vutukuru and Hari Balakrishnan
MIT CSAIL
{mythili,hari}@csail.mit.edu

Kyle Jamieson
University College London
k.jamieson@cs.ucl.ac.uk

ABSTRACT

This paper presents *SoftRate*, a wireless bit rate adaptation protocol that is responsive to rapidly varying channel conditions. Unlike previous work that uses either frame receptions or signal-to-noise ratio (SNR) estimates to select bit rates, SoftRate uses confidence information calculated by the physical layer and exported to higher layers via the *SoftPHY interface* to estimate the prevailing channel bit error rate (BER). Senders use this BER estimate, calculated over each received packet (even when the packet has no bit errors), to pick good bit rates. SoftRate's novel BER computation works across different wireless environments and hardware without requiring any retraining. SoftRate also uses abrupt changes in the BER estimate to identify interference, enabling it to reduce the bit rate only in response to channel errors caused by attenuation or fading. Our experiments conducted using a software radio prototype show that SoftRate achieves 2× higher throughput than popular frame-level protocols such as SampleRate [4] and RRAA [24]. It also achieves 20% more throughput than an SNR-based protocol trained on the operating environment, and up to 4× higher throughput than an untrained SNR-based protocol. The throughput gains using SoftRate stem from its ability to react to channel variations within a single packet-time and its robustness to collision losses.

Categories and Subject Descriptors: C.2.1 [Computer-Communication Networks]: Network Architecture and Design—Wireless communication

General Terms: Design, experimentation, performance.

Keywords: Wireless, bit rate adaptation, SoftPHY, cross-layer.

1. INTRODUCTION

Wireless communication suffers from many time-varying vagaries that cause bit errors and packet losses. These include signal *attenuation*, channel *fading* due to multipath propagation, and *interference* caused by other transmissions at overlapping frequencies. These stochastic effects are more pronounced when changes occur in the propagation environment, for instance because of node mobility, or by the movement of people and objects. The result is a channel that is difficult (if not near-impossible) to accurately

SIGCOMM'09, August 17–21, 2009, Barcelona, Spain.

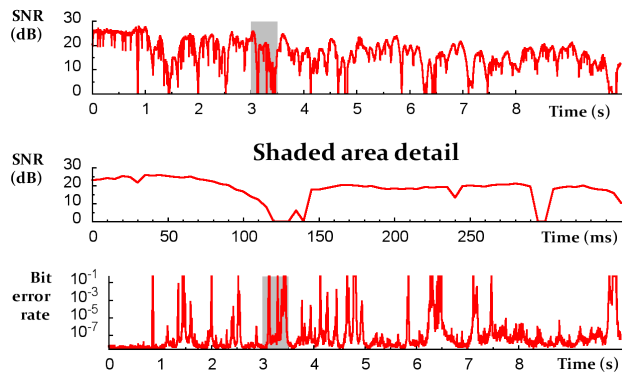


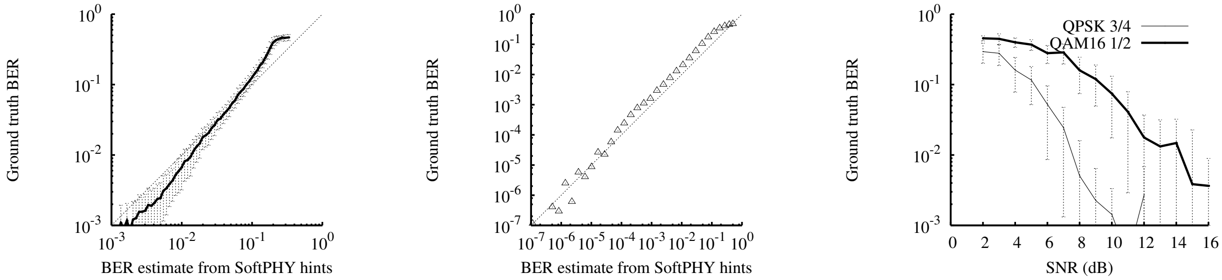
Figure 1: Experimental SNR fluctuations in time over a fading channel with walking-speed mobility. Large-scale fading is evident from the 10-second window (upper), and in a 350 ms detail (middle) we see fades a few tens of milliseconds in duration. Bit error rate (lower: BPSK, code rate-1/2) changes with SNR. Data obtained using an 802.11a/g-like software radio prototype (§4).

model, in which the signal-to-noise ratio (SNR) and channel bit error rate (BER) change with time. For example, Figure 1 shows measurements that illustrate the variation of SNR and BER over time when a sender is moving away from the receiver at walking speed; note the multipath fading effects on shorter timescales in addition to the gradual attenuation over longer timescales.

To improve throughput in these varying conditions, the sending node can dynamically adapt its modulation and coding by picking a suitable *bit rate*. The *bit rate adaptation protocol* used to make this choice must answer two important questions:

1. What signal (information) should the sender use to select the right bit rate?
2. Over what timescale should this signal be observed?

Prior work on bit rate adaptation (§2) uses one of two information signals: frame receptions or signal-to-noise ratio (SNR). Frame-level protocols [24, 4] must operate over the timescale of tens or hundreds of frames or more because they need several transmissions to accurately assess frame loss rates at various bit rates. As a result, frame-level schemes are not responsive to channel variations that occur on shorter timescales. On the other hand, SNR-based protocols [10, 21] can operate on shorter timescales by estimating the SNR on each reception and mapping it to the expected BER using known SNR-BER relationships. But because the BER at a given SNR might vary by many orders of magnitude between environments, these protocols must be carefully trained for each



(a) Per-frame average BER predicted by SoftPHY hints

(b) Aggregated average BER predicted by SoftPHY hints.

(c) Per-frame average BER predicted by SNR.

Figure 7: SoftPHY-based and SNR-based BER estimation in a static wireless channel.

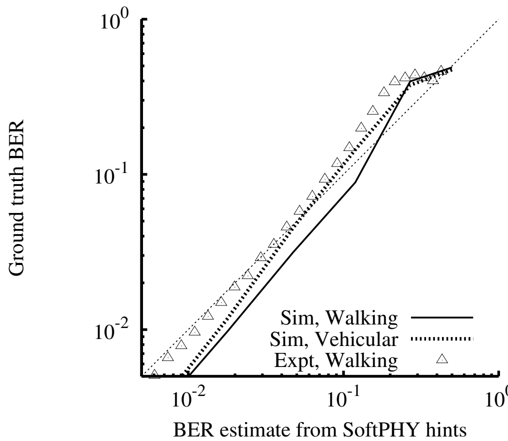


Figure 8: SoftPHY-based BER estimation in a mobile channel.

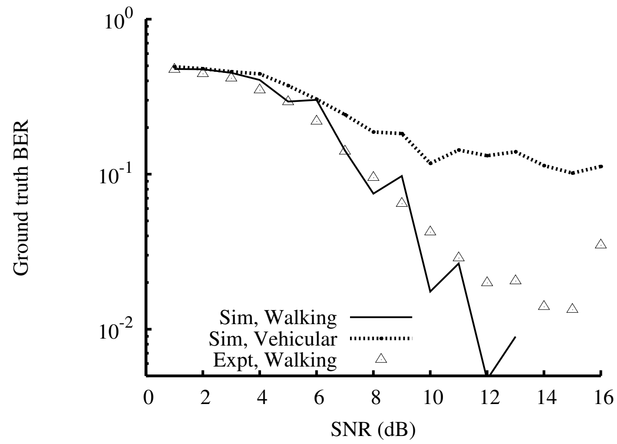


Figure 9: SNR-based BER estimation in a mobile channel.

for two bit rates, with the data binned as described earlier. We observe that a given SNR measurement corresponds to a very wide range of estimated BERs (the estimate has a mean error variance of 2.8×10^{-3} for QPSK 3/4 rate and 1.7×10^{-3} for QAM16 1/2 rate), illustrating that SNR is an unreliable predictor of BER.

BER prediction in mobile channels. We now show that SoftPHY hints reliably estimate BER even in mobile fading channels with widely varying channel coherence times. This section uses data from the walking and simulation traces of Table 4. For each dataset, we bin the data by SoftPHY-estimated BER, and compute the mean ground truth BER in each bin. Figure 8 shows the results, with the two curves corresponding to simulation traces at walking (Doppler spread 40 Hz) and vehicular speeds (Doppler spread 400 Hz), and the points in the figure corresponding to experimental data from the walking traces. Figure 9 shows the corresponding SNR-BER curves at the QAM16 1/2 rate.

From the figures, we see that the SoftPHY-based BER estimate is not sensitive to mobility speed while the SNR-BER curves are. The fact that the SNR-BER relationship changes with channel coherence time is well-known [19] and has also been observed experimentally by Camp and Knightly [5]. This happens because the SNR measured using the preamble does not capture the variation of SNR that happens over the body of the frame in fading channels, which in turn depends on the coherence time of the channel. On the other hand, SoftPHY hints reflect the increasing number of deep fades in the body of the frame as channel coherence time decreases, and therefore estimate BER across all wireless propagation environments accurately. Because the SNR-BER relationship changes with channel coherence time, SNR-based protocols must be carefully retrained for every operating environment; we show later (§6.3) that these protocols pick inaccurate bit rates and suffer a performance penalty if not retrained. In contrast, SoftRate can be used in any wireless propagation environment without retraining.

5.3 Interference Detection Accuracy

We now evaluate our implementation of our SoftPHY-based interference detection algorithm (as described in §3.2 and §4).

False positives. To measure the false positive rate (i.e., the rate at which the fading effects of the wireless channel are falsely identified as collisions), we collect the static and walking traces from Table 4 in a quiet frequency band without any other 802.11a/g transmissions. Out of the resulting frames lost, our collision detection algorithm identified less than 1% of them as collisions.

Interference detection accuracy. We use the traces from the static interference experiment described in Table 4 to measure the accuracy of our interference detection algorithm.

The sender-receiver link in the trace delivered 100% of its frames correctly in the absence of interference. In the presence of interference, one of three things can happen to a frame. First, the frame can be silently lost if the interferer transmits before the sender, ei-

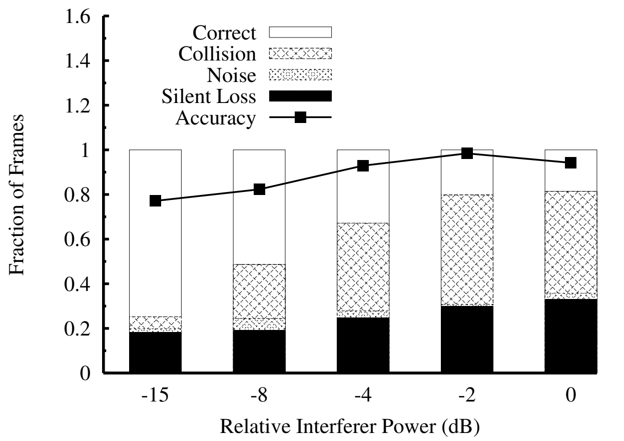


Figure 10: Interference detection accuracy as a function of varying interferer power.

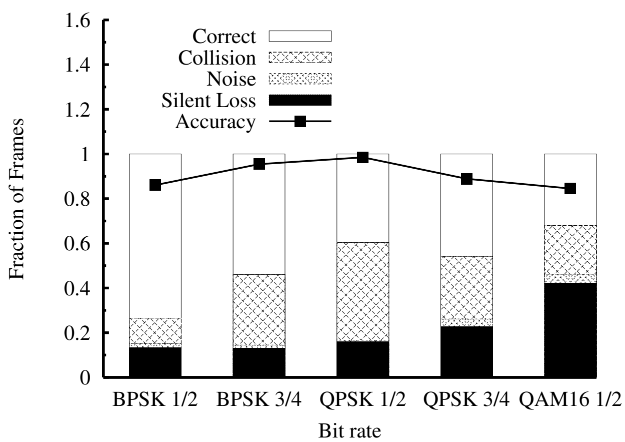


Figure 11: Interference detection accuracy as a function of transmit bit rate.

ther because the receiver has locked on to the interferer's frame, or because the sender's preamble is corrupted by the interferer's signal. Second, the frame can be received, but with errors. Finally, the frame can be correctly received. In the case of frames received with bit errors, we run our interference detection algorithm on the SoftPHY hint traces of the frame to see what fraction of these losses our algorithm identifies as collisions.[3]

We slice the interference detection accuracy results by the different transmit power levels of the interferer and the transmit bit rate of the sender. Figure 10 shows the fraction of frames that fall into each of the cases described above versus the relative interferer strength (in dB). Also shown on the graph is the interference detection accuracy of our algorithm, which is computed as the fraction of frames received with bit errors (i.e., the frames corresponding to "collision" and "noise" in the figure) that our algorithm correctly identifies as collisions. Figure 11 shows the same data, but broken down by the sender's bit rate. We find that our algorithm can always identify more than 80% of frames received in error as collisions. Because the colliding packets are of the same size in this

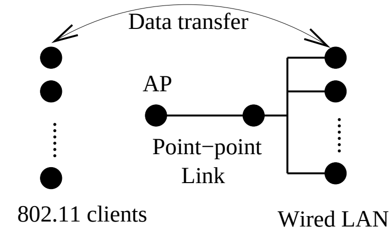


Figure 12: Topology used for the ns-3 evaluation of SoftRate.

experiment, we will be able to detect most of the silent losses as collisions as well by adding postambles.

6. EVALUATION OF SOFTRATE

In this section, we evaluate SoftRate using trace-driven simulations on ns-3, as described in §4. We quantify the performance gains for end-to-end TCP transfers when running SoftRate at the link layer in the following wireless environments: (1) Slow fading mobile channels, (2) Simulated fast fading channels, and (3) Interference-dominated channels.

We use TCP throughput as the metric to evaluate SoftRate against other rate adaptation protocols because applications like TCP and VOIP are more sensitive to losses, and therefore require responsive and accurate rate adaptation protocols to function well. While previous work mostly uses UDP throughput as a measure of performance, we believe that gains obtained on UDP transfers without congestion control are hard to realize in most practical applications.

6.1 Method

Trace-driven simulation. To conduct realistic simulations, we evaluate SoftRate using traces from software radio experiments described in Table 4. For each wireless link being simulated, we seed the simulator with a set of traces, one per bit rate, that completely specify the channel characteristics of the link (like, whether a frame sent is correctly received, and what its SNR and SoftPHY hints would be) for each point in time during the simulation. When the PHY in the simulator receives a frame at a certain bit rate, the fate of the frame is decided by looking up the appropriate trace. The bit rate adaptation protocol at the MAC layer receives and reacts to the feedback from the PHY (frame reception events, SNR estimates, or SoftPHY hints, as the case may be) and sets a suitable bit rate for the next frame. We make no assumptions on the symmetry of links, and use different traces for each of the two uni-directional links between every sender and receiver.

While collecting traces to be used in simulations, we ensure that the channel conditions are consistent across the various bit rates at any point of time. For traces collected using the channel simulator, we simulate the same fading process across experiments at different bit rates. We run live experiments in the short range mode with small frames sent at each of the bit rates in a round robin manner, running through all the bit rates once in under 5 milliseconds. We find that the BER across the various bit rates is monotonic in 96% of such 5 ms cycles, indicating that the channel is indeed fairly invariant across all the bit rates in a 5 ms snapshot.

All traces are collected with one sender transmitting at a time. In simulations with more than one sender, these traces collected in isolation accurately model frame receptions when there are no concurrent transmissions. In case more than two senders transmit simultaneously (e.g., experiments in interference-dominated channels in §6.4), we assume both colliding frames are lost.

[3] We omit here results for QAM16 3/4 rate, because our current implementation of that bit rate is untuned.

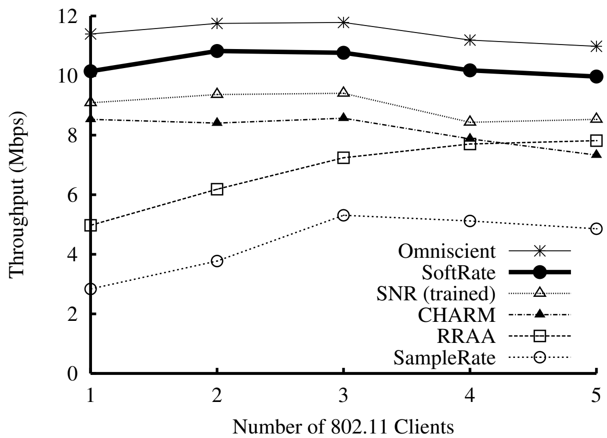


Figure 13: Aggregate TCP throughput (slow-fading mobility).

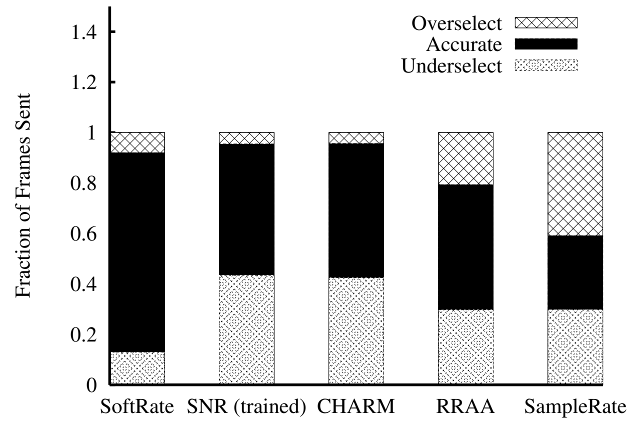


Figure 14: Rate selection accuracy with one TCP flow in a mobile slow fading channel.

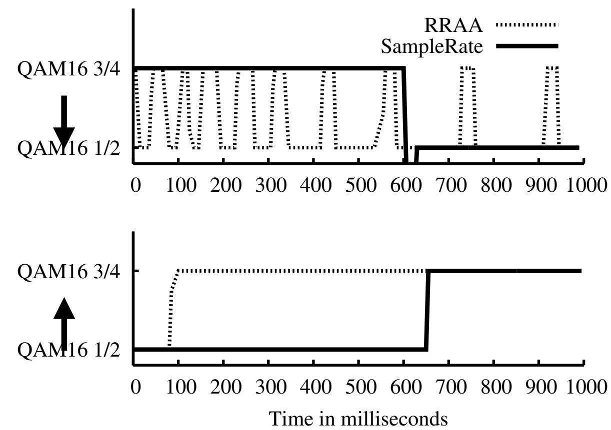


Figure 15: Bit rates chosen by RRAA and SampleRate where the optimal bit rate changes at $t = 0$: from a higher rate to a lower rate (top) and from lower to higher (bottom).

Simulation topology. The topology used in our simulations is shown in Figure 12. N clients connect to an access point (AP) that supports the 802.11a/g bit rates from 6 Mbps to 36 Mbps. The AP is connected to a LAN gateway node by a point-to-point link of bandwidth 50 Mbps and one-way delay of 10 ms. In each experiment, N TCP flows are set up to transfer 1400 byte data frames in either direction between the 802.11 clients and the corresponding wired LAN nodes. Each node's MAC queue length slightly exceeds the bandwidth-delay product of the bottleneck wireless link.

Algorithms evaluated. We compare the performance of SoftRate against the following rate adaptation algorithms.

1. Two SNR-based protocols: (i) a protocol that uses SNR feedback sent via the link-layer ACK to pick the transmit bit rate, much like RBAR but without the RTS/CTS overhead, and (ii) a protocol that uses the average SNR over multiple frames, much like CHARM[4]. The SNR-BER relationships for both protocols are computed from the traces used for evaluation.
2. Two frame-level schemes: (i) RRAA, and (ii) SampleRate. The various parameters in these protocols are set as described in the corresponding references, except for the interval over which transmission time averages are computed in SampleRate, for which a value of one second gave a better performance than the ten second value suggested in [4].
3. An "omniscient" algorithm that always picks the highest rate guaranteed to succeed, which a simulator with a priori knowledge of channel characteristics computes from the traces.

6.2 Slow Fading Mobile Channels

In this section, we evaluate how well SoftRate can adapt to channel variations that occur at walking speeds in a slow fading channel.

Simulation setup. We simulate $N = 1, \ldots 5$ TCP flows from the 802.11 clients to the corresponding wired LAN nodes. We use the ten walking traces (Table 4) to model the ten uni-directional links. We assume perfect carrier sense among all senders.

Results. Figure 13 shows the aggregate TCP throughput obtained by the various rate adaptation algorithms as a function of the number of flows. We find that SoftRate outperforms all other algorithms, and comes closest to the omniscient algorithm. SoftRate gets up to 20% higher throughput than both SNR-based algorithms trained over the traces because the BER prediction from SNR is noisier than that using SoftPHY hints. We also found that using averaged SNR information in CHARM leads to lower responsiveness to short-term SNR variations and hence slightly worse performance than using just the instantaneous SNR value. SoftRate achieves up to $2\times$ higher throughput than RRAA and almost $4\times$ higher throughput than SampleRate because frame-level algorithms cannot adapt fast enough to channel fades that are caused due to mobility, with the result that TCP ends up loosing multiple packets in a window and reduces its offered load. We find that the loss rate experienced by TCP is an order of magnitude higher with frame-level algorithms than it is with SoftRate. We repeat with clients receiving TCP traffic; results are similar to those described above.

For the simulation with one TCP flow, Figure 14 shows how the bit rates picked by the various algorithms on every transmitted frame compared against the highest bit rate that would have gotten the frame through at that time. We find that SoftRate chooses the correct bit rate over 80% of the time.

To better understand the performance of frame-level algorithms, we simulate RRAA and SampleRate using a synthetic trace, where the channel alternates between a "good" state (best transmit bit rate

[4]Our simulation does not need to rely on the channel reciprocity assumptions used in [13] because we can afford to change the 802.11 link-layer ACK in the simulator to piggyback SNR information, while CHARM aims to work with existing 802.11 cards.

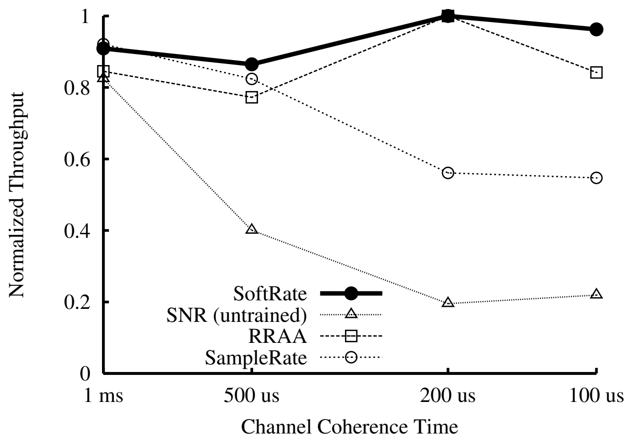


Figure 16: TCP throughput in a simulated fast fading channel.

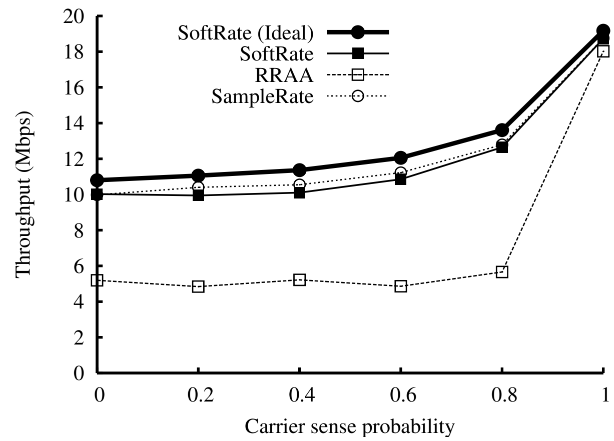


Figure 17: Aggregate TCP throughput as a function of carrier sense probability between the senders.

is QAM16 3/4) and a "bad" state (best transmit bit rate is QAM16 1/2) every 1 second. Frame trace data for the good and bad states are taken from appropriate snapshots in the walking trace described in Table 4. Figure 15 shows the bit rates picked by RRAA and SampleRate as a function of time, where the best transmit bit rate moves from the higher rate to the lower rate in the top panel, and back to higher rate in the bottom panel. The convergence times of RRAA and SampleRate are 15 ms and 600 ms respectively in the first case, and 85 ms and 650 ms in the second. These convergence times explain why the frame-level algorithms frequently overselect and underselect compared to the optimal in Figure 14. One other interesting point to note is the instability of RRAA's rate choice (see the top panel of Figure 15), highlighting another short-coming of frame-level algorithms. When the frame loss rate at a bit rate is zero, frame-level algorithms have no way of knowing if the frames are barely making it through (i.e., the next rate will not work) or if they are getting through very comfortably (i.e., next rate may work). SoftRate knows what the BER at the current rate is and hence can predict whether the next rate will work or not, obviating the need to unnecessarily probe higher rates.

Implications. Failing to adapt the transmit bit rate quickly to channel fades that occur with mobility can lead to burst losses that reduce TCP throughput. As a result, a responsive bit rate adaptation protocol like SoftRate offers huge gains for TCP in mobile channels, compared to less responsive frame-level algorithms.

6.3 Simulated Fast Fading Channels

In this section, we evaluate the performance of SoftRate in fast fading channels that occur at vehicular mobility speeds.

Simulation setup. One 802.11 client transfers TCP data to a wired LAN node via the AP. We use the simulation traces from Table 4 to model the links.

Results. We present the throughput of the various protocols normalized by the throughput of the omniscient algorithm because the best transmit bit rate (and hence the absolute throughput achieved) decreases with channel coherence time. Figure 16 shows the normalized throughput of the TCP flow with various rate adaptation algorithms as a function of varying channel coherence time. The SNR-BER relationships used by the SNR-based protocol are obtained over the walking traces used in §6.2. As channel coherence time reduces, the channel BER at any given bit rate increases for the same SNR. As a result, the SNR-based protocol underestimates the frame BER at lower coherence times and ends up selecting bit rates

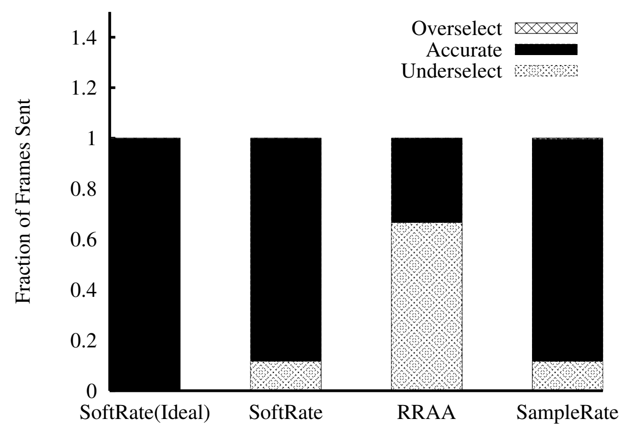


Figure 18: Rate selection accuracy ($\Pr[$carrier sense$] = 0.8$).

that are above optimal. Because SoftPHY hints measure the average BER over the entire frame, SoftRate correctly picks the bit rate that codes for the average channel BER in fast fading channels and its performance stays the same across various coherence times even without retraining. We see from the figure that SoftRate achieves a performance gain of about $4\times$ over the SNR-based protocol at a channel coherence time of 100 μs. Gains over CHARM were similar, as we did not use the retraining mechanism that adjusts SNR thresholds every few seconds, in order to isolate the impact of training on the performance of SNR-based protocols.

Implications. SNR-based protocols incur a performance penalty if not retrained for each operating environment, unlike SoftRate that works robustly across a wide variety of channel conditions without requiring retraining. CHARM proposes to retrain the SNR thresholds on a coarse timescale. However, such mechanisms are meant to handle calibration issues across different hardware and are ineffective if the coherence time of the channel changes on a short timescale, for example, when a train passes by a stationary user.

6.4 Interference-Dominated channels

In this section, we evaluate the impact of interference losses on the performance of SoftRate.

Simulation setup. The simulation consists of five 802.11 clients uploading TCP data via the AP to the wired LAN nodes. We use the static short range traces described in Table 4 to model each of the uni-directional links; using a static channel helps us isolate the benefits due to interference detection from those due to better adaptation in mobile channels. We simulate imperfect carrier sense between the various senders in the simulation to generate collisions. We vary the carrier sense probability between the senders from 0 (i.e., all senders are perfect hidden terminals) to 1 (i.e., perfect carrier sense and hence no interference losses). We simulate two versions of SoftRate–a present version where interference detection succeeds 80% of the time and there is no postamble detection, and a yet-to-be-implemented "ideal" version with postambles and perfect interference detection. When the SoftRate receiver identifies a frame loss as interference, the feedback BER from the receiver is simply the interference-free BER measured in the trace. Otherwise, the feedback is a very high BER indicating a noise loss.

Results. Figure 17 shows the performance of the various algorithms as a function of carrier sense probability. RRAA, because it reacts to short-term frame loss rate, reduces its bit rate in response to interference and sees a much lower throughput than the other algorithms. We found RRAA's Adaptive RTS/CTS scheme to be ineffective in preventing collisions, because interference was unpredictable and resulted in RTS/CTS being constantly turned on and off without any real benefits. SampleRate, on the other hand, is resilient to interference losses because it computes the average transmission time at each bit rate over slower timescales; interference affects the transmission time at all bit rates uniformly at such timescales. The performance of the omniscient algorithm is very similar to that of the ideal SoftRate and is not shown. The performance of the SNR-based algorithms is not affected by interference because the SNR was estimated using the preamble and not over the entire frame. Figure 18 shows the rates picked by the various algorithms on every transmitted frame, compared against the optimal bit rate choice. As expected, RRAA frequently underselects.

Implications. Algorithms that react to short-term channel variations entail the danger of lowering bit rate on interference losses. SoftRate's interference detection mechanism avoids this penalty.

7. CONCLUSION

We have presented SoftRate, a cross-layer wireless bit rate adaptation algorithm that achieves throughput gains of up to $2\times$ over frame-based protocols such as SampleRate and RRAA, 20% over SNR-based protocols trained on the operating environment, and $4\times$ over untrained SNR-based protocols. The key idea is to expose per-bit confidences called SoftPHY hints from the physical layer, using them to estimate the interference-free BER of received frames. Picking bit rates using the BER thus estimated enables SofTRate to react quickly to channel variation without requiring any environment-specific calibration. Moreover, SoftRate's idea of estimating BER from SoftPHY hints can be applied to a variety of wireless cross-layer protocols that, for example, allocate frequency or transmit power, or perform efficient error recovery.

Acknowledgments

We gratefully acknowledge the feedback we received from Brad Karp, our shepherd Ranveer Chandra, and anonymous reviewers. This work was supported in part by National Science Foundation grants CNS-0721702 and CNS-0520032 and by Foxconn.

8. REFERENCES

[1] P. A. K. Acharya, A. Sharma, E. M. Belding, K. C. Almeroth, and D. Papagiannaki. Congestion-Aware Rate Adaptation in Wireless Networks: A Measurement-Driven Approach. In *Proc. IEEE SECON Conf.*, pp. 1–9, San Francisco, CA, June 2008.

[2] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv. Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate (Corresp.). *IEEE Trans. on Information Theory*, 20(2):284–287, 1974.

[3] S. Biaz and N. H. Vaidya. Discriminating Congestion Losses from Wireless Losses Using Inter-arrival Times at the Receiver. In *Proc. of the IEEE ASSET Symp.*, pp. 10–17, Richardson, TX, Mar. 1999.

[4] J. Bicket. Bit-Rate Selection in Wireless Networks. Master's thesis, Massachusetts Institute of Technology, Feb. 2005.

[5] J. Camp and E. Knightly. Modulation Rate Adaptation in Urban and Vehicular Environments: Cross-Layer Implementation and Experimental Evaluation. In *Proc. of the ACM MobiCom Conf.*, pp. 315–326, San Francisco, CA, Sept. 2008.

[6] G. D. Forney, Jr. The Viterbi Algorithm (Invited Paper). *Proc. of the IEEE*, 61(3):268–278, Mar. 1973.

[7] S. Gollakota and D. Katabi. Zigzag Decoding: Combating Hidden Terminals in Wireless Networks. In *Proc. of the ACM SIGCOMM Conf.*, pp. 159–170, Seattle, WA, Aug. 2008.

[8] J. Hagenauer and P. Hoeher. A Viterbi Algorithm with Soft-Decision Outputs and its Applications. In *Proc. IEEE GLOBECOM*, pp. 1680–1686, Dallas, TX, Nov. 1989.

[9] D. Halperin, T. Anderson, and D. Wetherall. Taking the Sting out of Carrier Sense: Interference Canncelation for Wireless LANs. In *ACM MobiCom*, pp. 339–350, San Francisco, CA, Sept. 2008.

[10] G. Holland, N. Vaidya, and P. Bahl. A Rate-Adaptive MAC Protocol for Multihop Wireless Networks. In *Proc. of ACM MobiCom Conf.*, pp. 236–251, Rome, Italy, Sept. 2001.

[11] IEEE Standard 802.16e-2005: Air Interface for Fixed and Mobile Broadband Wireless Access Systems, Amendment 2, Feb. 2006. http://standards.ieee.org/getieee802/802.16.html.

[12] K. Jamieson and H. Balakrishnan. PPR: Partial Packet Recovery for Wireless Networks. In *Proc. ACM SIGCOMM*, pp. 409–420, Kyoto, Japan, August 2007.

[13] G. Judd, X. Wang, and P. Steenkiste. Efficient Channel-aware Rate Adaptation in Dynamic Environments. In *Proc. of the ACM MobiSys Conf.*, pp. 118–131, Breckenridge, CO, June 2008.

[14] A. Kamerman and L. Monteban. WaveLAN II: a High-Performance Wireless LAN for the Unlicensed Band. *Bell Labs Technical Journal*, 2(3):118–133, Summer 1997.

[15] K. C. Lin, N. Kushman, and D. Katabi. ZipTx: Exploiting the Gap Between Bit Errors and Packet Loss. In *Proc. of the ACM MobiCom Conf.*, pp. 351–362, San Francisco, CA, Sept. 2008.

[16] D. Mandelbaum. An Adaptive-Feedback Coding Scheme Using Incremental Redundancy (Corresp.). *IEEE Trans. on Information Theory*, 20(3):388–389, May 1974.

[17] J. Metzner. Improvements in Block-Retransmission Schemes. *IEEE Trans. on Communications*, 27(2):524–532, Feb. 1979.

[18] ONOE Rate Control. http://madwifi.org/browser/trunk/ath_rate/onoe.

[19] J. G. Proakis. *Digital Communications, 4th ed.* McGraw-Hill, 2000.

[20] S. Rayanchu, A. Mishra, D. Agrawal, S. Saha, and S. Banerjee. Diagnosing Wireless Packet Losses in 802.11: Separating Collision from Weak Signal. In *Proc. of IEEE INFOCOM Conf.*, pp. 735–743, Phoenix, AZ, Apr. 2008.

[21] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly. Opportunistic Media Access for Multirate Ad Hoc Networks. In *Proc. of ACM MobiCom Conf.*, pp. 24–35, Atlanta, GA, Sept. 2002.

[22] T. M. Schmidl and D. C. Cox. Robust Frequency and Timing Synchroniation for OFDM. *IEEE Trans. on Communications.*, 45:1613–1621, Dec. 1997.

[23] D. Tse and P. Viswanath. *Fundamentals of Wireless Communication*. Cambridge Univ. Press, 2005.

[24] S. Wong, H. Yang, S. Lu, and V. Bharghavan. Robust Rate Adaptation for 802.11 Wireless Networks. In *Proc. of ACM MobiCom Conf.*, pp. 146–157, Los Angeles, CA, Sept. 2006.

[25] J. Zhang, K. Tan, J. Zhao, H. Wu, and Y. Zhang. A Practical SNR-Guided Rate Adaptation. In *Proc. of the IEEE INFOCOM Conf.*, pp. 2083–2091, Phoenix, AZ, Apr. 2008.

[26] Y. Zheng and C. Xiao. Simulation Models With Correct Statistical Properties for Rayleigh Fading Channels. *IEEE Trans. on Communications*, 51(6):920–928, 2003.