

Harnessing Frequency Diversity in Wi-Fi Networks

Apurv Bhartia Yi-Chao Chen Swati Rallapalli Lili Qiu
The University of Texas at Austin
{apurvb,yichao,swati,lili}@cs.utexas.edu

ABSTRACT

Wireless multicarrier communication systems transmit data by spreading it over multiple subcarriers and are widely used today owing to their robustness to multipath fading, high spectrum efficiency, and ease of implementation. In this paper, we use real measurements to show that there is significant frequency diversity in Wi-Fi channels, and propose a series of techniques to explicitly harness such frequency diversity. In particular, we leverage the Channel State Information (CSI), which captures the SNR on each subcarrier to (i) map symbols to subcarriers according to their importance, (ii) effectively recover partially corrupted FEC groups and facilitate FEC decoding, and (iii) develop MAC-layer FEC to offer different degrees of protection to the symbols according to their importance and error rates at the PHY layer. We further develop a rate adaptation approach that works together with these optimization schemes. Our trace-driven simulation and testbed experiments on USRP clearly demonstrate the effectiveness of our approaches.

Categories and Subject Descriptors

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

General Terms

Algorithms, Experimentation, Measurement, Performance

Keywords

IEEE 802.11, Forward Error Correction (FEC), Orthogonal Frequency Division Multiplexing (OFDM), Cross Layer Design, Rate Adaptation.

1. INTRODUCTION

Wireless multicarrier communication systems, such as Orthogonal Frequency Division Multiplexing (OFDM), transmit data by spreading it over multiple orthogonal subcarriers. Since each individual subcarrier is narrowband and has slow symbol rate, it is easy to eliminate inter-symbol interference arising from frequency selective fading. Owing to its robustness to multipath fading, high spectrum efficiency, and ease of implementation, OFDM is widely

used in many wireless systems, such as in digital radio and television broadcast, wireless local area networks (*e.g.*, IEEE 802.11a/g/n), personal area networks (*e.g.*, UWB), and metropolitan area networks (*e.g.*, WiMax and LTE).

Due to frequency selective fading and frequency dependent attenuation, it is common for different subcarriers to experience different signal quality [8, 9, 24]. Such signal variation across different frequencies can have fundamental impacts on multicarrier network performance.

In this paper, we first measure and analyze Channel State Information (CSI) of a Wi-Fi channel, which captures the signal to noise ratio (SNR) on each subcarrier. We observe that there exists significant frequency diversity and the SNR across different subcarriers sometimes differs by 10 dB or more. In addition, we find that CSI is fairly stable and can be predicted with high accuracy using the Holt-Winters forecast [15]. Moreover, the communication cost of a receiver's reporting the CSI to the sender is small. Therefore it is feasible for a sender to leverage the CSI to optimize its transmission and increase throughput.

Motivated by these observations, we propose a series of techniques to explicitly harness such frequency diversity:

- We first develop smart symbol interleaving schemes, which map important symbols to more reliable subcarriers for transmission in order to maximize the total throughput.
- We then leverage CSI information as hints to perform partial FEC group recovery and facilitate FEC decoding.
- We further develop a novel MAC-layer FEC scheme to maximize the throughput by offering different degrees of protection to the symbols that experience different error rates at the PHY-layer and have different importance.
- Finally, we observe that our techniques can effectively reduce the perceived loss rate and potentially support a higher data rate. Therefore we develop a rate adaptation scheme that works together with the above optimization techniques.

We evaluate our approaches using both trace-driven simulation and testbed experiments in USRP [29]. Our results clearly demonstrate the benefit of our approaches. In particular, our results show that the smart symbol mapping, CSI-based hints, and MAC-layer FEC can each improve the throughput by 120% to 1580%, 26% to 91%, and 7% to 207%, respectively. Moreover, there exists synergy between these techniques, and leveraging them together yields higher throughput gain. In addition, our rate adaptation can effectively take advantage of these techniques, and improve throughput by 33% to 147% over the traditional rate adaptation.

The rest of the paper is organized as follows. We analyze the CSI traces collected in static and mobile networks in Section 2. We

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MobiCom'11, September 19–23, 2011, Las Vegas, Nevada, USA.
Copyright 2011 ACM 978-1-4503-0492-4/11/09 ...\$10.00.

present our approach in Section 3. We describe our trace-driven simulation methodology and results in Section 4. We present our implementation and testbed results in Section 5. We survey the related work in Section 6. We conclude in Section 7.

2. ANALYSIS OF CSI TRACES

Background: According to the IEEE 802.11n standard, the network interface cards (NICs) report a standard Channel State Information (CSI). The CSI is a collection of $M \times N$ matrices H_s , each of which specifies amplitude and phase between pairs of N transmit and M receive antennas on subcarrier s . SNR and amplitude A have the following relationship: $SNR = 10\log_{10}(A^2/N)$, where N denotes the average power of white noise. We use Intel Wi-Fi Link 5300 (iwl5300) IEEE a/b/g/n wireless network adapters to collect the CSI of each frame preamble across all subcarriers. These NICs have three antennas. We enable all three antennas at the receivers and one antenna at the sender. The modified driver [12] reports the channel matrices for 30 subcarrier groups, which is about one group for every two subcarriers in a 20 MHz channel according to the standard [2] (*i.e.*, 4 groups have one subcarrier each, and the other 26 groups have two subcarriers each). All the measurements are conducted in 5 GHz channel 36 to avoid interference with the campus Wi-Fi networks. We use 1000-byte packets, MCS 0, a transmission power of 15 dBm. MCS 0 has 1 stream, so the NICs report CSI in the form of 1×3 matrices for each frame.

Trace collection: We collected CSI traces on the sixth floor of the ACES building at The University of Texas at Austin. It is a regular office building with student cubicles across the floor. Every node is a desktop equipped with the Intel Wi-Fi Link 5300 (iwl5300) adapter. For the static traces, we let a sender placed at one cubicle broadcast traffic, and let five receivers spread across different cubicles record the CSI of the received frames for an hour. Each trace contains about 2,500,000 packets. We use the first 100,000 packets in all of our analysis. In addition, we collected a mobile trace by having one sender broadcast to four receivers placed at different cubicles. We walked with the sender at an approximate speed of 10 meters in half a minute (*i.e.*, about 1.2 kmph). Each mobile trace contains around 20,000 packets and we use all the packets for our analysis. Throughout this paper, we refer to the five static traces as traces 1 to 5, and refer to the four mobile traces as traces 6 to 9.

Frequency diversity: We first use the CSI measurements to show that there exists significant frequency diversity. For each received frame, we compute the difference between minimum and maximum SNR across different subcarriers in a 20 MHz 802.11 channel. Figure 1 plots the cumulative distribution (CDF) of the difference over 5 sender-receiver pairs in the static traces. Each node pair consists of three links because there are three antennas at the receivers, effectively giving us 15 links in total. We make the following observations. First, all the links exhibit significant signal variation across different frequencies. Several links often see over 10 dB difference across different frequencies. Second, as we would expect, the degree of frequency diversity varies across the links: the difference between the minimum and maximum SNR varies from 3.6 to 5.7 dB on some links, and from 9.8 to 32.4 dB on other links.

Next we analyze the mobile traces, and plot the CDF of the difference between maximum and minimum SNR across different subcarriers in Figure 2. There are effectively $4 \times 3 = 12$ measured links. As before, we observe that all links exhibit significant frequency variation, and the difference between maximum and minimum SNR even for the link with minimum variation is more than 8.3 dB for 50% of the packets.

Note that the frequency diversity exists not only in single an-

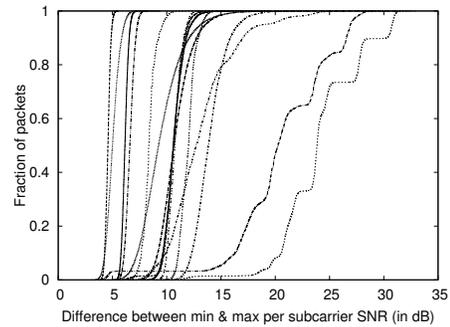


Figure 1: Heterogeneity in the signals across different frequencies within a 20 MHz 802.11 static channel.

tenna cases (as shown in the above analysis results) but also in multiple antenna cases. For example, the frequency diversity seen by a MIMO receiver is the same as the diversity on its individual antennas when it uses spatial multiplexing since each antenna is responsible for receiving one stream. These measurement results motivate us to develop techniques to exploit such frequency diversity in Wi-Fi networks.

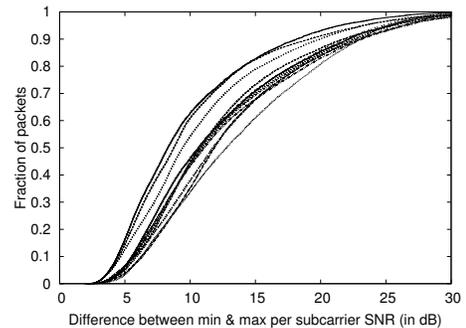


Figure 2: Heterogeneity in the signals across different frequencies within a 20 MHz 802.11 mobile channel.

Temporal stability: In order to effectively exploit such frequency diversity, the CSI should be relatively stable over time so that we can predict the future CSI using the past measurements. We quantify the temporal stability by computing the normalized amplitude change, defined as follows:

$$\frac{\|A(t1) - A(t2)\|_2}{\|A(t2)\|_2},$$

where $A(t1)$ and $A(t2)$ are the amplitudes of the two consecutive packets on all the subcarriers, and $\|\cdot\|_2$ is the ℓ_2 -norm (with $\|\vec{z}\|_2 = \sqrt{\sum_k \vec{z}(k)^2}$ for any vector \vec{z}). Figure 3 plots the CDF of normalized amplitude change for the static traces. As we can see, for most traces, over 94% of cases amplitude changes within 10% between two consecutive packets.

Figure 4 plots the CDF of the amplitude change between two consecutive packets in the mobile traces. As we see, the amplitude changes within 20% in 92.1% of cases.

We further study the temporal stability as we vary the gap between the packets. Figure 5 plots CDF of the relative difference between the CSI of the two packets with sequence numbers $t1$ and $t1 + gap$, *i.e.*,

$$\frac{\|A(t1) - A(t1 + gap)\|_2}{\|A(t1 + gap)\|_2},$$

where gap varies from 1 to 10. In the static trace, the difference is similar with different gaps between the two packets, which suggests

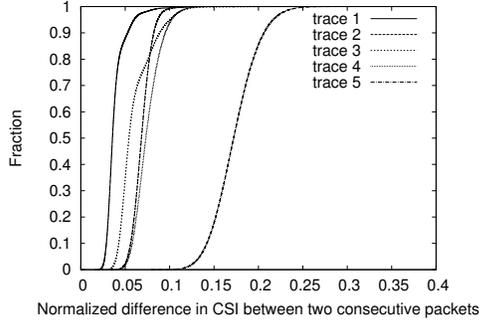


Figure 3: Temporal stability of CSI in the static traces.

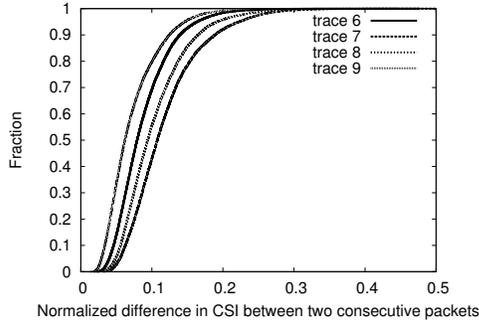


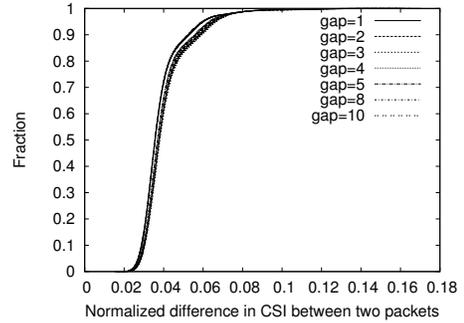
Figure 4: Temporal stability of the CSI in the mobile traces.

that CSI fluctuates randomly. In the mobile trace, the difference increases with the gap; but even when the gap increases to 10, the difference is still within 16% for 50% of the frames. We observe similar results for the other traces. The significant temporal stability in the CSI suggests that it is possible for us to predict the future CSI using the historical measurements and optimize performance based on predicted CSI.

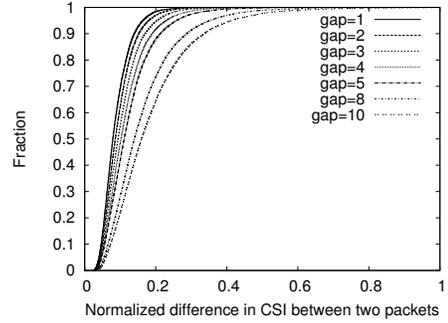
CSI prediction: Next we examine the prediction algorithms for CSI. We observe that in the static traces the CSI fluctuation tends to be random, whereas in the mobile traces the CSI has a linear trend (e.g., when the sender is moving away from or towards the receivers) in addition to random fluctuation. Therefore, in general, the CSI series consist of two components: a baseline and a linear trend. To account for these two components, we use the Holt-Winters forecasting algorithm [15] to predict CSI. Let y denote the CSI time series on a given subcarrier. y consists of a baseline component a and a linear trend component b , both of which are estimated using exponential weighted moving average (EWMA). Then the future sample, denoted as $y(i+1)$, can be predicted as follows, where α and β are the weights we put on the most recent sample. Our evaluation uses $\alpha = 0.2$ and $\beta = 0.1$.

$$\begin{aligned}
 a(i) &= \alpha \times y(i) + (1 - \alpha) \times (a(i-1) + b(i-1)) \\
 b(i) &= \beta \times (a(i) - a(i-1)) + (1 - \beta) \times b(i-1) \\
 y(i+1) &= a(i) + b(i)
 \end{aligned}$$

Figure 6 plots the prediction error in static and mobile traces. For comparison, we also plot the prediction error of EWMA, which estimates the future sample $y(i+1)$ as $\alpha x(i) + (1 - \alpha)y(i)$, where $x(i)$ is the actual value of i -th sample and $y(i)$ is the predicted value of i -th sample. As we would expect, for EWMA, small α performs better in the static traces, and large α performs better in the mobile traces. Therefore choosing an appropriate α a priori in EWMA is difficult. In comparison, the Holt-Winters forecasting with the same α and β performs comparably to or better than

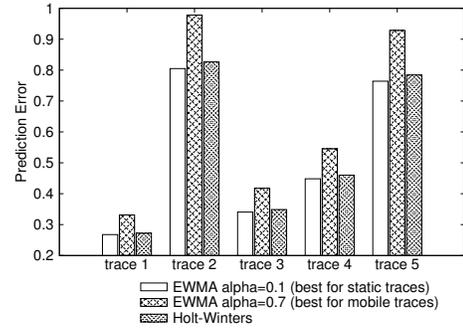


(a) Static: trace 1

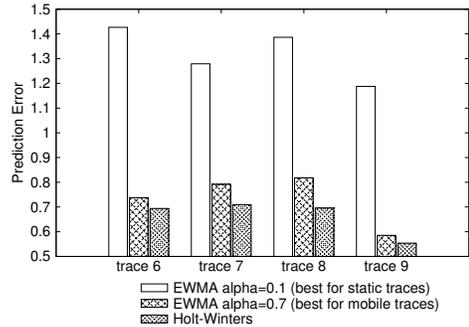


(b) Mobile: trace 6

Figure 5: Changes in CSI with a varying time gap.



(a) Static traces



(b) Mobile traces

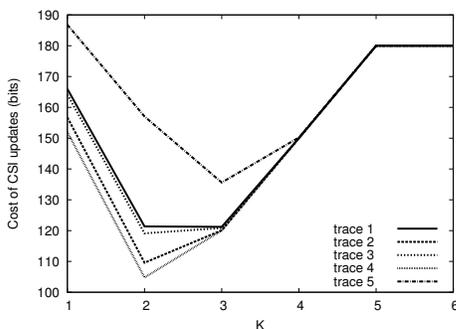
Figure 6: Prediction error of CSI in both the static and mobile traces.

the EWMA scheme using the best configuration on both static and mobile traces. This is because it decomposes the time series into two components and separately keeps track of the baseline and the linear trend using the same parameters. Instead of changing the parameters, the linear trend component captures the degree of the

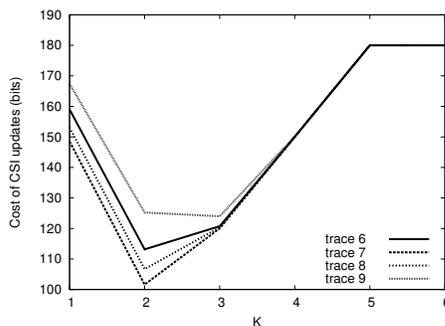
mobility and increases with mobility. Due to the effectiveness and generality of the Holt-Winters algorithm, we use it for prediction throughout the rest of this paper.

Cost of updating CSI information: A receiver needs to feedback CSI information to its sender so that the sender can use it to optimize its strategy for future transmissions. Simply sending the raw CSI information (*e.g.*, amplitude) for every frame can be costly. To reduce overhead, it is natural to use delta compression to send the amplitude change between two consecutive packets. We find the maximum amplitude change from one packet to the next is well below 64 and can be safely represented using 6 bits for each sub-carrier. Since the driver reports CSI as a vector of size 30 from each antenna, the standard delta compression requires $30 \times 6 = 180$ bits. We use the following enhancement to further reduce the update cost. When the change is within K , we send it using $\log_2(K)$ bits; when the change is above K , we send the change using 6 bits. We use a binary indicator to denote which mode we use for each subcarrier (*i.e.*, whether using $\log_2(K)$ or 6 bits).

Figure 7 shows the average update cost for every packet across the entire static and mobile traces as we vary K . The update cost includes the cost of the indicators. As it shows, the cost is lowest when $K = 2$ or 3. At these values, the CSI is around 100-120 bits per antenna, which is easily affordable. For example, even at the lowest data rate of one antenna (6.5 Mbps), the ACK time increases within 24%; at the highest data rate of one stream (65 Mbps), the ACK increases within 4.5%. Since ACK transmission time is a small portion of the data transmission time, this overhead is within 1% even at the lowest data rate for 1500-byte data frames. Furthermore, we find that decreasing the CSI update frequency by half has negligible impact on the performance due to significant stability. This suggests that CSI information can be available at low cost so that we can use it to optimize future transmissions.



(a) Static traces



(b) Mobile traces

Figure 7: Cost of updating CSI in the static and mobile traces.

3. OUR APPROACH

Motivated by the significant frequency diversity within a channel, we develop a series of techniques to harness such diversity and improve throughput.

3.1 Smart Symbol Interleaving

Standard interleaver: According to IEEE 802.11 standard, all data bits are interleaved by a block interleaver with a block size corresponding to the number of bits in one OFDM symbol [11, 1]. Let N_{CBPS} denote the block size. The block interleaver is a two-step permutation procedure. The first permutation step maps adjacent coded bits to non-adjacent subcarriers. The second permutation maps adjacent coded bits alternatively to less and more significant bits of the constellation to avoid long runs of low reliability bits. Let k denote the index of the coded bit before the first permutation, i denote the index after the first permutation, and j denote the index after the second permutation. The two permutation steps are defined as follows, where $k = 0, 1, \dots, N_{CBPS} - 1$, $i = 0, 1, \dots, N_{CBPS} - 1$, and s is determined by the number of coded bits per subcarrier (N_{BPSC}) according to $s = \max(N_{BPSC}/2, 1)$.

$$i = (N_{CBPS}/16)(k \bmod 16) + \text{floor}(k/16) \quad (1)$$

$$j = s \times \text{floor}(i/s) + (i + N_{CBPS} - \text{floor}(16 \times i/N_{CBPS})) \bmod s \quad (2)$$

Smart interleavers: The goal of the standard interleaver is to avoid long runs of low reliability bits by spreading adjacent bits apart to reduce correlated losses. However, they do not consider that different subcarriers can experience very different loss rates and different symbols can have different importance. Smart interleavers incorporate both frequency diversity and importance of symbols. Here are a few examples where different bits within a frame may have different importance:

- A packet header is more important than payload.
- When a systematic code is used (*e.g.*, Reed-Solomon code, systematic convolutional code, and standard LDPC), the original data symbols are transmitted along with redundancy symbols and the former are more important because they can be decoded once they are received correctly, whereas the latter are only useful when enough symbols in the FEC group are received. For example, consider a systematic FEC group that consists of 5 data symbols and 5 redundancy symbols. Correct reception of 4 data symbols alone leads to 4 symbols worth of throughput (when we use CSI as hints in Section 3.2), whereas correct reception of 4 redundancy symbols alone is useless.
- When transmitting an image or video, symbols corresponding to the main object in the scene are more important than the symbols pertaining to the surrounding background.

Based on this observation, we propose to map important symbols to reliable subcarriers. Our design focuses on interleaving for the first two scenarios, though our general techniques can be easily extended to other contexts. In order to support such mapping schemes, we require cross layer information (*e.g.*, location of the header in a frame). This can be achieved by having the upper layer mark the bits in a frame that require higher level of protection.

The idea of mapping important symbols to reliable subcarriers is natural, but the challenge lies in how to map symbols to subcarriers in order to maximize the total throughput. Before delving into the detailed approaches, we first formally specify the problem. Given a set of subcarriers and the loss rate of each subcarrier, we want to

determine the symbol to subcarrier mapping that maximizes the expected received payload, defined as $P_{header} \times N$, where P_{header} is the delivery rate of a header and N is expected number of delivered payload data symbols. The mapping should specify for each FEC group, which subcarriers are used to transmit data symbols for the header, data symbols for the payload, and redundancy symbols for the header and payload.

Due to the non-linear utility function, finding the optimal solution to this mapping problem is challenging. We develop several heuristics. Our first three heuristics are based on the insights that header symbols and data symbols are generally more important.

- **Smart header:** It divides symbols into two groups: header symbols and payload symbols. It maps header symbols to the most reliable subcarriers and maps payload symbols to the remaining subcarriers.
- **Smart data:** It divides symbols into two groups: data symbols and FEC redundancy symbols. It maps data symbols to the most reliable subcarriers and maps redundancy symbols to the remaining subcarriers.
- **Smart header/data:** It sorts the subcarriers in an increasing order of their SNR, and partitions them into four groups: (i) data symbols in the header, (ii) data symbols in the payload, (iii) redundancy symbols in the header, and (iv) redundancy symbols in the payload. Let n_1, n_2, n_3, n_4 denote their sizes in a frame, respectively. Then we map (i) to the best n_1 subcarriers in terms of SNR, map (ii) to the next best n_2 subcarriers, map (iii) to the following top n_3 subcarriers, and map (iv) to the remaining subcarriers.

Since the above three heuristics do not provide optimal performance, we develop the following smart iterative search to further improve the performance. We first use the above three smart mapping schemes to derive three initial mappings. Then, for each initial mapping, we try to iteratively improve upon it as follows. We swap K symbols from one FEC group with another FEC group. If the swap leads to an increase in the utility, we accept it; otherwise, we ignore it. In both cases, we iterate until the maximum number of iterations is reached. Our evaluation uses 5 iterations. Finally, among the three refined mappings, we pick the one that yields the best performance.

It is important to determine which two FEC groups to swap. Our intuition is that allocating reliable subcarriers to an FEC group with low delivery rate can be more beneficial than allocating them to an already highly reliable FEC group. Therefore, during each iteration, we sort the FEC groups in the order of their delivery rates and randomly swap the subcarriers between the best and worst FEC groups in terms of their delivery rates. One caveat is that when the best FEC group involves a header, swapping its subcarriers with less reliable ones may degrade the performance due to a higher header corruption rate. Therefore, in that case, we compute the utility of two scenarios: either swap the group involving the header with the worst FEC group or swap the next best FEC group (which does not contain a header) with the worst FEC group. We will take the swapping that gives a higher utility.

In all four smart interleaving schemes, to avoid long runs of erroneous bits within each of these partitions (which contains the symbols of the same priority), we apply a similar permutation step as Equation 1 to spread the adjacent bits apart. In this way, we not only provide prioritized treatments to important bits but also avoid long runs of low reliability bits.

Receiver feedback: To allow a receiver to properly decode a frame, the receiver has to know the exact interleaving used at the sender.

This can be either explicitly transmitted by the sender or implicitly computed by the receiver. We take the latter approach due to its lower communication overhead. The receiver reports the CSI information in the ACK. Based on the CSI information, the sender and receiver will run the same interleaving algorithm and use the same permutation step so that they should arrive at the same interleaving. The only information that the sender needs to communicate is which CSI the sender uses to compute the interleaving. This is simply the sequence number of the ACK, which includes the CSI used for computing the interleaving. Such sequence number could be represented using a small number of bits without ambiguity.

3.2 Leveraging Channel Information for FEC decoding

Next we propose to use CSI to facilitate decoding in the following two ways: (i) help recover the data symbols when its FEC group cannot be completely decoded, and (ii) improve FEC recovery capability by knowing the likely error positions.

Recovering partial FEC groups: Specifically, if an FEC group fails (*i.e.*, whose errors exceeds a threshold), the current decoder cannot extract any symbols from the failed group. In comparison, we can use SNR of the subcarrier, on which the symbols are transmitted, and extract the data symbols whose SNR exceeds a threshold. In our evaluation, we set the SNR threshold to correspond to 0.1% BER under the current modulation scheme. In this way, we can perform partial FEC group recovery. The benefit of using CSI over existing approaches (*e.g.*, SoftPHY [14]) is that CSI is available even in commodity hardware, which we can immediately benefit, whereas SoftPHY hints are only available in software-defined radios. For those bits that fail the CSI check, we may request for retransmissions as PPR [14]. The frame is considered successfully delivered only if it passes the frame CRC check after retransmission(s). In this way, we do not compromise integrity of the frame.

Improving FEC group recovery: In addition to partial recovery, CSI information is useful to increasing the FEC group recovery probability. For example, a LDPC code is widely used in wireless systems, such as IEEE 802.11n, WiMax, and LTE. It uses the channel information to decode an FEC group. In a nutshell, LDPC uses several parity checks to perform error detection and recovery. When the received symbols in an FEC group fail one or multiple of the parity checks, the LDPC decoder will flip symbols within the group according to the error probability of the symbols and then use parity bits to check the consistency of the new FEC group. The flips will continue until the symbols pass all the parity checks in that group or the number of flips exceed a threshold. An accurate estimation of the error probability for the symbols is important to the decoding success. As shown in Section 4.2, using the error probability calculated based on the CSI information can considerably increase the LDPC decoding success rate over the current practice, which assumes uniform error probability. Given the increasing popularity of LDPC codes, our enhancement has significant practical value. Moreover, such benefit can potentially also extend to other FEC schemes, *e.g.*, Reed Solomon, where we can also flip bits according to their error probability in a Reed Solomon FEC group until they pass the consistency check. The main difference is that LDPC checks consistency using parity checks, which is much more efficient than the consistency check in Reed Solomon. As a result, the latter is applicable only to a wireless network with a relatively low data rate.

3.3 MAC-layer FEC

Due to frequency diversity, there may not exist a single PHY-layer data rate that works best for all the subcarriers. One way

to handle such heterogeneous subcarrier quality is to use different modulation and PHY-layer FEC on each subcarrier (*e.g.*, as in [24]). However, this may require mapping the symbols within a PHY FEC group onto the same subcarrier, since different subcarriers may simply use different FEC codes. This requirement can lead to undesirable bursty losses. Moreover, commodity hardware (*e.g.*, Wi-Fi cards) does not have the flexibility in controlling modulation at the per-subcarrier level, and can only select the data rate for the entire frame. Furthermore, frequency-aware rate adaptation also introduces significant signaling and processing overhead [27].

We propose to use MAC-layer FEC to offer different degrees of protection to effectively maximize the total throughput. MAC-layer FEC has several advantages: (i) it is flexible and can offer different levels of protection to different symbols based on their importance and PHY-layer loss rates, (ii) it is more fine-grained than PHY-layer FEC: it allows us to use not only different FEC group sizes but also any FEC redundancy level instead of being constrained by the limited choices offered at the PHY layer, *e.g.*, IEEE 802.11a offers only 1/2 (half data symbols in an FEC group), 3/4 (3/4 data symbols in an FEC group), and 2/3 FEC while IEEE 802.11n offers 1/2, 3/4, 2/3, 5/6 FEC at the PHY-layer; (iii) it can be easily deployed on commodity hardware.

Problem formulation: Our goal is to maximize the throughput by selectively adding MAC-layer FEC. The approach can be easily extended to offer differentiated treatment to FEC groups if certain groups are more important than others (*e.g.*, header vs. payload). Here we consider that the data rate and bit interleaving information are given. In Section 3.4, we will relax this assumption and jointly optimize rate, bit interleaving, partial recovery, and MAC-layer FEC. We consider Reed Solomon FEC code at the MAC layer. It consists of K data symbols and $N - K$ redundancy symbols. A group is correctly received if there are no more than $(N - K)/2$ incorrect symbols.

This problem has a large search space: (i) how much MAC-layer FEC to add, (ii) how to split these FEC symbols to protect different PHY-layer symbols, and (iii) what FEC group size to use at the MAC layer. To limit the search space, we partition the PHY-layer symbols in each FEC group into two groups: good symbols with low BER and bad symbols with high BER. Let d_g and d_b denote the number of good and bad PHY-layer symbols, respectively. Our problem is to decide how to split the PHY-layer symbols into these two groups and how many MAC-layer FEC symbols to allocate to protect each of these two groups. Let r denote the total number of MAC-layer FEC symbols we will add. Let r_g and r_b denote the number of MAC-layer FEC symbols allocated to protect good and bad PHY symbols, respectively. Obviously, $r = r_g + r_b$. Our goal is to find (d_b, r_b, r_g) that maximizes throughput.

Algorithm: To achieve this goal, for every FEC group in a frame, we search for the best combination of (d_b, r_b, r_g) such that it maximizes the effective delivery rate, defined as the expected number of delivered symbols (N_r) divided by the total number of symbols transmitted (N_t), which includes MAC-layer FEC symbols. Figure 8 shows the pseudo-code. The transmitter will then add the MAC-layer FEC for each PHY-layer FEC according to $config_{max}$ as calculated. In this case, the sender does not need to notify the receiver of the MAC-layer FEC it uses. This is because both the sender and the receiver operate on the same CSI information and run the same algorithm which is deterministic in nature. This avoids communication overhead that might otherwise be needed. Furthermore, to minimize the correlated losses within one MAC-layer FEC group, we use the hash function in Equation 1 to map adjacent FEC symbols to different subcarriers. Since the sender

```

foreach PHY-layer FEC group
  foreach  $(d_b, r_b, r_g)$ 
    compute  $R_g = N_r/N_t$ 
    if  $R_g > R_g^{max}$ 
       $R_g^{max} = R$ 
       $config_g^{max} = (d_b, r_b, r_g)$ 
    end
  end
end
end

```

Figure 8: Pseudo code of MAC-layer FEC.

and receiver use the same hash function, no additional information about the mapping needs to be transmitted.

The algorithm maximizes the utility N_r/N_t for each FEC group. N_t is simply $d + r_g + r_b$. Below we describe how to compute N_r . For a given (d_b, r_b, r_g) , we estimate the expected number of successfully delivered data symbols without partial FEC recovery (*i.e.*, the technique described in Section 3.2) as follows:

$$N_r = P1 \times d1 + (1 - P1) \times P2_g \times d_g + (1 - P1) \times P2_b \times d_b \quad (3)$$

where $P1$ denotes the delivery probability of the FEC group using PHY-layer FEC, $P2_g$ and $P2_b$ are the delivery probabilities of the FEC group using MAC-layer FEC for the good and bad FEC groups, respectively, and $d1$ is the number of data symbols in the group. The expression assumes independence between delivery probabilities of PHY-layer FEC and MAC-layer FEC. The group is delivered when either of these FEC groups succeeds.

When partial FEC recovery is enabled as described in Section 3.2, the expected number of successfully delivered data symbols increases to the following:

$$N_r = P1 \times d1 + (1 - P1) \times P2_g \times d_g + (1 - P1) \times P2_b \times d_b + (1 - P1) \times (1 - P2) \times \left(\sum_{i \in bad} P3_i + \sum_{i \in good} P3_i \right) \quad (4)$$

where $P3_i$ is the delivery probability of the i -th symbol that belongs to either a good or bad group. The expression reflects the fact that a symbol is received if (i) its PHY-layer FEC group is correctly received, or (ii) its MAC-layer FEC group is correctly received (but its PHY-layer FEC group is incorrect), or (iii) the symbol itself is correctly received (but its PHY-layer and MAC-layer FEC groups are both incorrect). Among these variables, $P3_i$ can be easily computed from the CSI using SNR to BER mapping. Below we focus on computing $P1$ and $P2$, both of which are the delivery probability of one FEC group.

Computing the delivery probability of one FEC group: The delivery probability of one FEC group is essentially the probability of the FEC group having no more than T errors, where T is the maximum number of errors that the FEC group can tolerate. This can be approximated using a normal distribution. Since an FEC group typically contains around or over 100 symbols, the approximation using a normal distribution works well according to the law of large numbers. This is also confirmed in our evaluation, which shows its estimation is quite close to the actual delivery rate.

Specifically, let BER_i denote the BER of the subcarrier i and I is the set of subcarriers used to carry symbols in the current FEC group. $X(i)$ denotes if the bit i is corrupted and define $Y = \sum_{i \in I} X(i)$, which denotes the expected number of corrupted bits in the FEC group. We can approximate Y as $N(\mu, \sigma)$, where $\mu = \sum_{i \in I} BER_i$ and $\sigma = \sqrt{\sum_{i \in I} BER_i(1 - BER_i)}$. Then the delivery probability of an FEC group can be approximated as

$\theta(\frac{T-\mu}{\sigma})$, where $\theta(x)$ is the CDF of the standard normal distribution $N(0, 1)$.

3.4 Combining with Rate Adaptation

The above three optimization techniques reduce the perceived loss rate and can potentially support a higher data rate. Motivated by this observation, below we first describe how to jointly optimize rate adaptation with smart symbol interleaving and then show how it can be extended to also leverage partial FEC group recovery and MAC-layer FEC.

Joint rate adaptation and smart interleaving: To jointly optimize rate adaptation and smart symbol interleaving, we enumerate all data rates, and for each data rate we identify the best symbol interleaving based on the CSI information. Then we select the data rate and symbol mapping that maximize the total throughput.

To achieve this goal, we need to accurately estimate the throughput for a given symbol interleaving and data rate. We compute throughput as the expected number of correctly received symbols in the payload data divided by the transmission time $TxTime$, which includes DIFS, SIFS, header and ACK overhead. The denominator is easy to compute, so we focus on computing the numerator. We observe that a data symbol s is successfully received if (i) the header is correctly received and (ii) the FEC group to which s belongs is received correctly. Therefore, we first compute the delivery probability of an individual FEC group, and then compute the expected number of received symbols as follows:

$$P_{header} \cdot \sum_K P_{FEC(K)} d_K,$$

where P_{header} denotes the header delivery probability, $P_{FEC(K)}$ denotes the FEC group K 's delivery probability, and d_K denotes the number of data symbols in the FEC group K .

P_{header} can be computed as follows:

$$\prod_{i \in header} BER_i + (1 - \prod_{i \in header} BER_i) P_{FEC'},$$

since the header is received either when all bits in the header are correctly received (which can be verified using a header CRC) or when the FEC group to which the header belongs is correctly received, which is denoted by $P_{FEC'}$. We can compute the probability of an FEC group having within T errors (e.g., $P_{FEC(K)}$ and $P_{FEC'}$), by approximating it using a normal distribution, as described in Section 3.3.

Joint rate adaptation, smart interleaving, and partial FEC recovery: To further incorporate partial group recovery based on the CSI hints, we simply redefine the throughput by including the partially correct symbols in the FEC group K as follows:

$$N_r = (P_{header} \cdot \sum_K (P_{FEC(K)} d_K + (1 - P_{FEC(K)}) \sum_{i \in K} (1 - BER_i))) \quad (5)$$

$$Throughput = N_r / TxTime \quad (6)$$

where the term $(1 - P_{FEC(K)}) \sum_{i \in K} (1 - BER_i)$ denotes the total number of partially correct symbols in the FEC group K . The intuition is that a data symbol s is received if (i) the header is received correctly, and (ii) the FEC group to which s belongs is received correctly or s is received correctly.

Joint rate adaptation, smart interleaving, partial FEC recovery, and MAC FEC: We extend our algorithm in Figure 8 to Figure 9 to support rate adaptation, smart interleaving, partial FEC recovery, and MAC FEC. It searches over all data rates, and for

```

foreach data rate
  use smart mapping to select a good symbol mapping
  foreach PHY-layer FEC group
    foreach  $(d_b, r_b, r_g)$ 
      compute  $R_g = N_r / N_t$ 
      if  $R_g > R_g^{max}$ 
         $R_g^{max} = R$ 
         $config_g^{max} = (d_b, r_b, r_g)$ 
      end
    end
  end
  compute  $TxTime$  under the current rate and MAC FEC
   $utility = P_{header} \times \sum_g R_g^{max} / TxTime;$ 
  if  $utility > max\_utility$ 
     $max\_utility = utility;$ 
    record the current rate and  $config_g^{max}$ 
  end
end

```

Figure 9: Pseudo code of MAC-layer FEC.

each data rate it finds the symbol mapping and MAC-layer FEC that maximizes the total throughput. The new utility effectively captures the impact of data rate, interleaver, partially correct FEC groups, and MAC-layer FEC overhead. It finally selects the combination of rate, symbol mapping, MAC-layer FEC that yields the highest throughput. For efficiency, we do not have to go through all the data rates during the search, and can prune a data rate if its throughput under no losses is no larger than the highest throughput found so far.

4. SIMULATION

4.1 Simulation Methodology

We conduct extensive trace driven simulations to study the benefit of our approaches. We collect CSI traces as described in Section 2 using Intel Wi-Fi Link 5300 (iwl5300) IEEE 802.11 a/b/g/n wireless network adapters on desktops. These NICs have three antennas and we use the trace from the first antenna in the following results since we observe similar results when using the other two antennas. For the static traces, we use the first 20,000 packets from each trace for simulation, and these packets correspond to about half a minute transfer. For the mobile traces, we only have about 20,000 packets per trace, so we use all the packets for simulation.

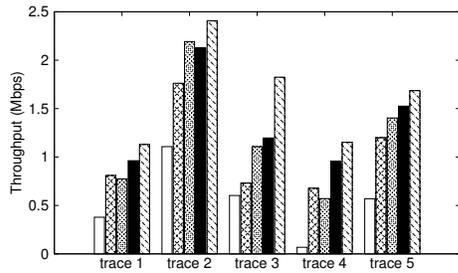
We develop an event-driven simulator as follows. A sender continuously transmits frames to a receiver. The symbols within a frame are mapped to subcarriers according to a given mapping scheme. The corruption rate of a symbol is determined according to the SNR of its subcarrier and the standard function that maps SNR to bit-error-rate (BER), as shown in Table 2 of [9] and also validated empirically in that paper. The sender optimizes its strategy, such as symbol mapping and MAC-layer FEC, according to the predicted CSI using the Holt-Winters forecasting algorithm. We use throughput as our performance metric. Throughput is defined as the total number of correctly received bits divided by the total time, which includes the transmission time and random backoff time. When CSI is used to extract partially correct symbols, the throughput only counts the symbols that are both correct and have CSI above the threshold.

In order to quantify the benefit of any loss recovery schemes, we have to use traces with packet losses. However, CSI information of lost packets are simply not available since the driver may not even know the presence of a packet. Therefore we collect the CSI traces

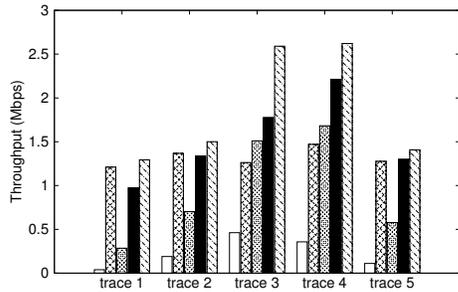
under good link conditions where it has little or no packet losses. Then we shift the original CSI as reported by the traces by a constant across all the subcarriers to emulate the effect that the sender uses a lower transmission power or is farther away so that the link experiences losses. The shift is constant across all the subcarriers so that it preserves the frequency diversity in the original traces. We identify the shift based on the modulation scheme so that the modulation scheme is appropriate for the current shift value.

We evaluate the performance under fixed rate and auto-rate. We use the static traces to evaluate the fixed rate, and use both the static and mobile traces to evaluate the performance of auto-rate. We do not use the mobile traces to evaluate the fixed rate case since the SNR varies significantly and it does not make sense to use a fixed rate throughout the entire duration of mobile traces.

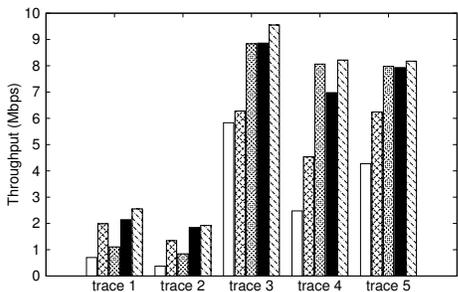
4.2 Performance Results



(a) BPSK, 1/2 FEC



(b) QPSK, 1/2 FEC

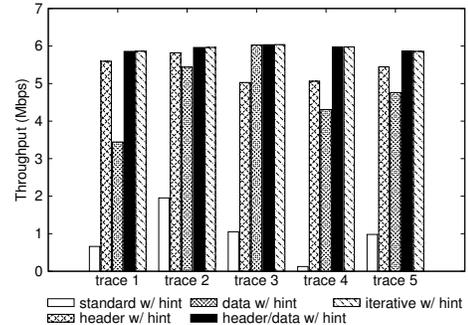


(c) QPSK, 3/4 FEC

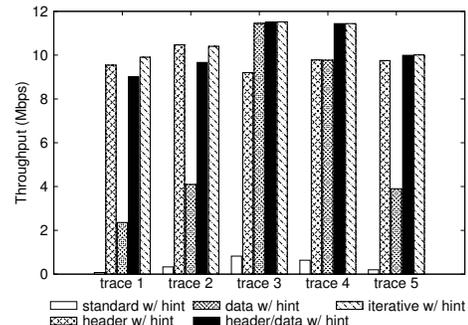
Figure 10: Comparison of different symbol mapping schemes under fixed rates in the static trace.

Evaluation of symbol mappings: Figure 10 compares the performance of our four smart mapping schemes with the standard mapping used in IEEE 802.11 [1, 11]. As shown in Figure 10, all

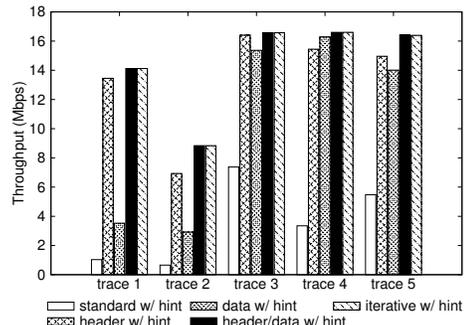
four smart mapping schemes consistently out-perform the standard mapping. The improvement of smart iterative search over the standard mapping ranges from 120% to 1580% in BPSK 1/2 FEC (*i.e.*, half of the symbols in an FEC group are data symbols), 460% to 3190% in QPSK 1/2 FEC, and 63% to 410% in QPSK 3/4 FEC (*i.e.*, 3/4 of the symbols in an FEC group are data symbols). Comparing the four smart mapping schemes, we see that smart iterative search consistently performs the best.



(a) BPSK, 1/2 FEC



(b) QPSK, 1/2 FEC



(c) QPSK, 3/4 FEC

Figure 11: Comparison of CSI-based hints under fixed rates in the static trace.

Evaluation of CSI-based hints: Figure 11 evaluates all the mapping schemes with partial recovery using CSI as hints. Comparing Figure 10 and Figure 11, we see that the CSI-based hints significantly improve the throughput of all mapping schemes. In BPSK 1/2 FEC, it improves throughput from 73% to 84% in the standard mapping, from 230% to 750% in the smart header mapping, from 150% to 650% in the smart data mapping, from 180% to 530% in the smart header/data mapping, and from 150% to 420% in the smart iterative search. The benefits of CSI hints are higher in the smart mapping schemes than those in the standard mapping for the following reason. The smart mappings allocate data symbols to

more reliable subcarriers so that they are more likely to be correct and whenever they are correct, the CSI hints allow us to extract the data symbols, which directly contributes to the throughput. In comparison, applying the CSI hints to the standard mapping yields lower throughput because while the CSI hints continue to allow us to extract correct symbols but a larger number of these symbols are redundancy symbols, which by themselves are not useful unless the FEC group receives enough correct symbols.

In addition, we observe that the smart iterative search with CSI hints improves the throughput of the default scheme (*i.e.*, the standard mapping without CSI hints) by 440% to 8630%, which is higher than the individual technique alone. This demonstrates the synergy between these techniques.

Next we use CSI to estimate the error probability in LDPC. Our result based on the static traces shows that this allows an LDPC using 273-bit codeword and 191-bit redundancy to successfully decode 20.2% more FEC groups than the current practice, which assumes uniform error probability.

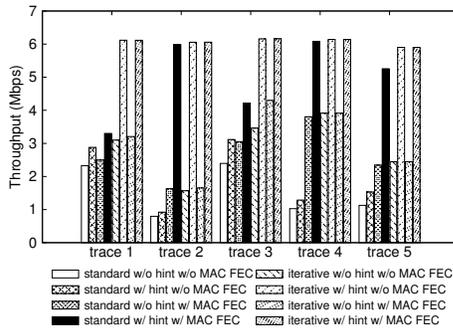


Figure 12: Evaluation of MAC FEC under the fixed rate (BPSK, 1/2 FEC) in the static traces.

Evaluation of MAC FEC: Figure 12 compares the performance with and without MAC FEC. As we can see, in the standard mapping, MAC FEC improves the total throughput by 7% to 207% when CSI are not used as hints, and improves throughput by 15% to 549% when CSI are used as hints.

Evaluation of the joint optimization: As shown in Figure 12, MAC FEC with smart iterative mapping and CSI hints improves the throughput of the default scheme (*i.e.*, no MAC FEC, no CSI hints, and standard mapping) by 160% to 660% , even higher than the benefit of individual technique alone, which confirms that the techniques work well together.

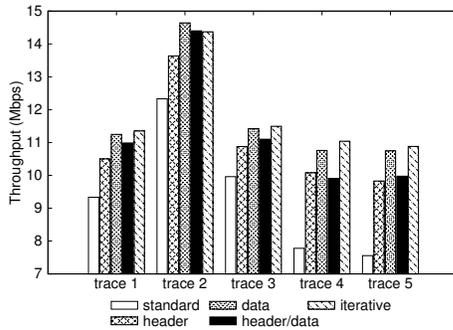


Figure 13: Comparison of different symbol mapping schemes under auto rates in the static traces.

Effects of our rate adaptation: We further evaluate the effectiveness of jointly optimizing data rate and symbol mapping. We compare rate adaptation using the smart mapping schemes against the

standard mapping. In both cases, we use the same auto-rate adaptation as described in Section 3.4, which selects the rate that maximizes throughput. They only differ in the way in which the symbols are mapped. As shown in Figure 13, jointly optimizing rate adaptation and symbol mapping out-performs the standard mapping with auto-rate. The throughput benefit ranges from 15% to 44% in the smart iterative search.

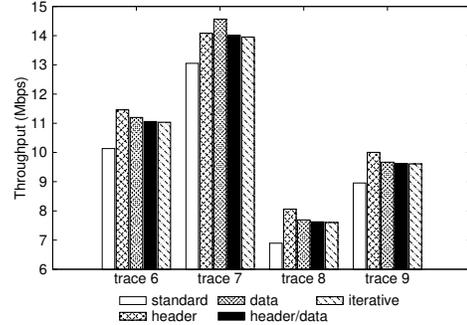


Figure 14: Comparison of different symbol mapping schemes under auto rates in the mobile traces.

Next we evaluate the performance of different mapping schemes under auto-rate on the mobile trace. This is more challenging since the prediction accuracy degrades under mobility and we may not accurately identify good subcarriers. As shown in Figure 14, all four smart mapping schemes continue to improve the throughput in the mobile traces. The improvement over the standard mapping ranges between 6.9% to 10.3% in the smart iterative search.

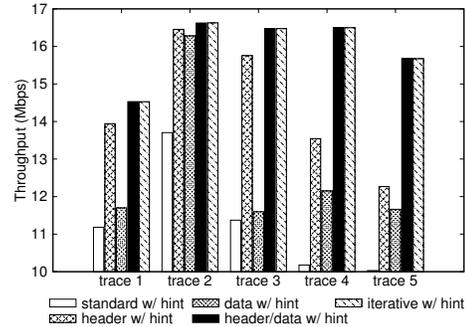


Figure 15: Comparison of CSI-based hints under auto-rate in the static traces.

Now we evaluate the throughput under auto rate with and without CSI hints. Comparing Figure 13 and Figure 15, we observe that the CSI hints in the auto-rate algorithm improve the throughput from 11% to 33% in the standard mapping, from 21% to 45% in the smart header mapping, from 2% to 13% in the smart data mapping, from 15% to 67% in the smart header/data mapping, and from 16% to 49% in the smart iterative search. The CSI hints and smart iterative search together improve the throughput of the default auto-rate scheme by 35% to 112%, which demonstrates the effectiveness of simultaneously leveraging these techniques.

Figure 16 shows the performance in mobile traces. Comparing Figure 14 and Figure 16, we see that the benefit of CSI-based hints extends to the mobile scenarios. CSI hints with auto-rate increase throughput from 5.3% to 5.8% in the standard mapping, and from 56.6% to 77.8% in the smart iterative search. The CSI hints with smart iterative search together improve the throughput by 68% to 96%, which confirms the effectiveness of joint optimization.

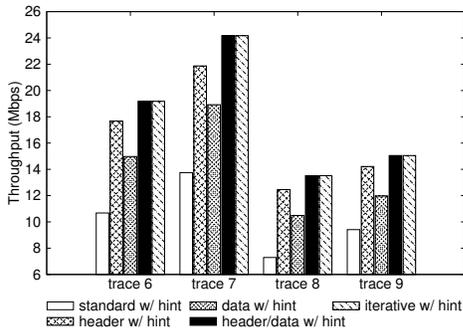


Figure 16: Comparison of CSI-based hints under auto-rate in the mobile traces.

5. TESTBED EXPERIMENTS

5.1 Implementation

As our USRP1 can only support up to 800 KHz, we see more homogeneous channels than 20-MHz Wi-Fi channels. This is an artifact of USRP1. As newer generation of software defined radios, such as USRP2 and SORA, support wider bands, they should see more frequency diversity. For our experiments, in order to recreate a similar degree of frequency diversity as we see in a Wi-Fi channel, we use three USRP nodes where two of them communicate using various schemes and the third USRP injects narrowband interference at the boundary of the channel used by the first two USRP nodes. In this way, the subcarriers on the boundary have lower SNR while the other subcarriers have higher SNR. We vary the interference by changing its transmission power and channel width. In our experiments, the difference between maximum and minimum SNR across different subcarriers are comparable to the Wi-Fi channel. In each transmission, we send out 1000 packets, each having 1000 bytes. Every packet has a CRC-protected header followed by the CRC-protected payload. The entire packet is divided into blocks, each of which is encoded using Reed Solomon FEC. To minimize the change in channel conditions, for each run, we run different schemes back to back on the USRP. For each experiment, we perform five different runs and report the total throughput for each run. Note that the throughput in different figures are obtained under different channel conditions and cannot be directly compared.

We implement our approaches on top of the standard OFDM examples in the GNU Radio 3.2.2 package. The default OFDM implementation in this package includes a basic OFDM sender and receiver chain. It supports BPSK, QPSK and QAM modulation schemes on the OFDM subcarriers. Our implementation uses 192 subcarriers and operates in the 2.49 GHz range.

We implement the standard mapping and three smart mapping schemes. In addition, we can enable or disable CSI hints for partial FEC group recovery. We use QAM-8 and QAM-16 as the modulation schemes in the following experiments.

5.2 Performance Results

Evaluation of symbol mappings: We plot the performance of different symbol mapping schemes in Figure 17 for five runs, each with different amount of narrowband interference. The throughput is calculated based on the number of correctly received FEC groups when the packet header is correctly decoded. When the header is corrupted, the entire packet is considered to be lost. We observe that all smart mapping schemes consistently perform better than the standard interleaver. For instance, the smart header/data scheme out-performs the standard interleaver by 42% to 173%, be-

cause it maps important header and data symbols to more reliable subcarriers and allows more FEC groups to be decoded.

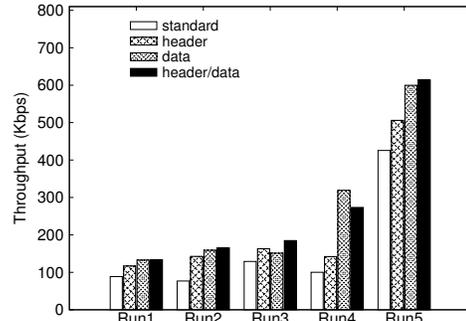


Figure 17: Comparison of different symbol mapping schemes in USRP.

Evaluation of CSI-based hints: We also use the CSI-based hints to extract the correct bits from the FEC groups that cannot be decoded at the receiver. For each such FEC group, we consider a bit to be correct if (1) it is indeed correct based on the ground-truth value of the bit and (2) the BER of the subcarrier on which the bit is received is within 0.005. Again we use the standard function to map SNR to BER. Figure 18 plots the resulting throughput after partial recovery for the various schemes during 5 separate runs. We observe that all the smart interleaver schemes consistently outperform the standard interleaver.

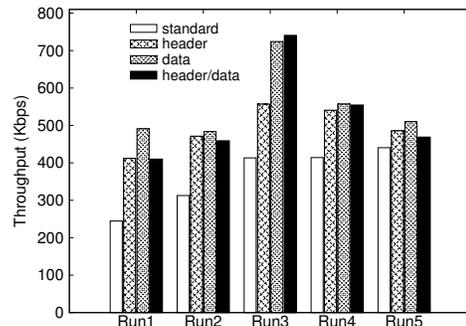


Figure 18: Comparison of CSI-based hints in USRP.

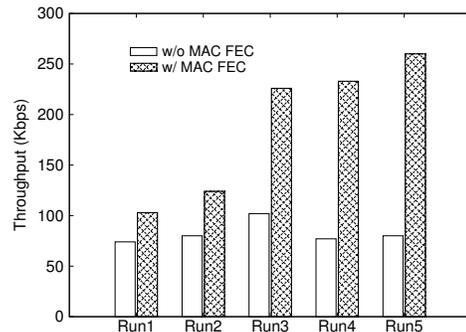


Figure 19: Evaluation of MAC-layer FEC in USRP.

Evaluation of MAC-layer FEC: We also evaluate the impact of MAC-layer FEC on USRP. Figure 19 shows the performance under 5 different scenarios with different loss rates. We can see that MAC-layer FEC improves throughput by 40% to 230% in the standard mapping.

Evaluation of the joint optimization: Figure 20 compares the performance of jointly applying MAC-layer FEC, smart header/data

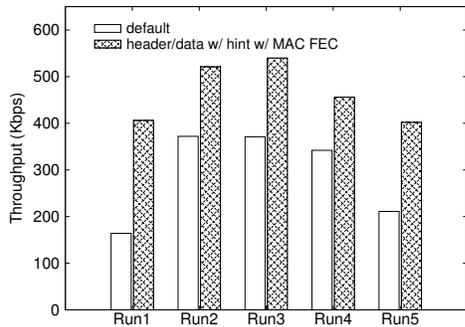


Figure 20: Evaluation of MAC-layer FEC with smart mapping in USRP.

mapping, and CSI hints and the default scheme, which does not use any of these techniques. As we can see, the combined approach out-performs the default scheme by 33% to 147%.

6. RELATED WORK

Our research is related to the works on frequency diversity, rate adaptation, and partial packet recovery, which we overview below.

Frequency diversity: Wired cable modem, such as DSL, uses frequency-aware modulation across OFDM subbands [28]. More recently, the wireless research community also recognizes the importance of frequency diversity in wideband transmissions (*e.g.*, 100 MHz channel) and has started developing a range of techniques to harness such diversity. For example, FARA [24] develops frequency-aware rate adaptation and MAC protocol. Our work differs from FARA by exploiting the frequency diversity using new techniques, such as mapping symbols to subcarriers and leveraging channel information to improve delivery rate. Our third technique, MAC-layer FEC, is inspired by frequency-aware rate adaptation but differs from it in that (i) it can offer more fine-grained choices of FEC instead of being restricted to a small number of modulation/FEC configurations at the physical layer, and (ii) using two layers of FEC coding is more effective in combating long error bursts than the current PHY-layer FEC, which uses short convolution codes and cannot handle long error bursts. Moreover, our approach can be easily applied to commodity hardware while FARA is only applicable to software-defined radios. On the other hand, FARA-like approaches can benefit from more efficient hardware implementation. It is possible to combine FARA with our PHY/MAC FEC so that we can benefit from frequency-aware modulation at the PHY layer and also leverage more fine-grained MAC-layer FEC to achieve higher efficiency.

[9] shows that frequency diversity is also present in 20-MHz Wi-Fi channels. They propose the notion of effective SNR to accurately estimate the loss rate based on CSI and develop a simple rate adaptation scheme based on CSI. Our work is motivated by [9] and our techniques exploit the frequency diversity in several new dimensions and we further develop a joint rate adaptation that works together with these techniques to achieve higher efficiency. [19] exploits the frequency diversity in retransmissions. In particular, it remaps the symbols in the retransmissions to avoid the same set of symbols getting corrupted and is especially useful under collisions. In comparison, our techniques are more general and apply to any transmissions, including both original transmissions and retransmissions, and reduce the need of retransmissions. Moreover, our approaches are useful with and without interference.

The third generation partnership project (3GPP) Long Term Evolution (LTE) exploits the frequency diversity by designing time-

frequency domain scheduling algorithms. A variety of scheduling techniques have been proposed in the literature (*e.g.*, [3, 18, 33]). Our techniques explore new control strategies to harness frequency diversity. They are complimentary and can potentially be used in conjunction with these scheduling schemes.

Rate adaptation: Rate adaptation is an extensively studied topic, and various rate adaptation algorithms have been proposed in the literature [4, 10, 16, 17, 22, 23, 25, 26, 31]. For example, [22] is the rate adaptation algorithm used in the MadWiFi driver. It uses long-term loss rate estimation and threshold to determine rate changes. SampleRate [4], proposed by Bicket et al., probes the performance at a random rate every 10 frames, and selects the rate that minimizes expected transmission time including retransmission time. Wong et al. [31] identify the limitations of existing design guidelines for rate adaptation. Based on their observations, they develop Robust Rate Adaptation Algorithm (RRAA), which uses short-term loss ratio to opportunistically change rate and incorporates an adaptive RTS filter to prevent collision losses from reducing data rate. All these existing rate adaptation schemes adapt rate according to frame loss rate. When partial packet recovery is used, the frame loss rate overestimates the actual loss rate experienced by data traffic and causes an unnecessarily low transmission rate to be used. [13, 20] suggest adaptation of rates according to the final loss rate after going through partial packet recovery. Our rate adaptation work together with smart mapping, hints, and MAC-layer FEC to achieve even a higher benefit.

[30] and [5] both propose adapting data rates according to the fine-grained BER information instead of coarse-grained frame error rate (FER). The former uses the physical layer hints to obtain BER while the latter develops an error estimation code to compute BER. [26] proposes a rate adaptation scheme that operates on symbol constellation and directly computes the optimal rate for the previous packet based on the dispersion between the transmitted and received symbol positions. These three approaches implicitly assume that SNR or symbol dispersion is the same throughout the frame. Due to frequency diversity, the SNR and symbol dispersion can vary significantly within a frame. The combination of PHY-layer and MAC-layer FEC, as in our approach, can more effectively handle the signal quality variation within a frame because we do not have to use a low physical rate on the entire frame just because a few symbols experience poor quality and can leverage MAC-layer FEC to protect these symbols.

Partial packet recovery: Partial packet recovery has received significant research attention recently. [7] uses fragment-based CRC so that correctly received fragments can be extracted even though the entire frame is not received correctly. [14] proposes to use the physical layer hints to indicate the confidence of received symbols and only the symbols with low confidence need to be retransmitted. [21] combines portions of the frames received from multiple radios for recovery. [32] leverages both physical layer hints and multiple radios to more effectively combine the portions of the frames from different radios. Unlike [21, 32], which work for multiple radios, our approach works for both single and multiple radios. Our CSI-based hints are related to the physical layer hints, but unlike physical layer hints, which is only available in software defined radio, CSI information is available in commodity hardware so that we can benefit immediately. As our evaluation shows that there is synergy between smart mapping and hints, leveraging both allows us to achieve even better performance than using the hints alone. Moreover, we not only use CSI-based hints to extract correct symbols from partially correct frames but also use it to increase the likelihood for LDPC code to successfully decode the FEC group.

7. CONCLUSION

In this paper, we use measurements from IEEE 802.11 networks to analyze the extent of the frequency diversity, its temporal stability, and update cost. Our traces show that CSI exhibits strong diversity and temporal locality, which make it possible for us to effectively harness such diversity. We develop three complementary techniques to effectively harness such diversity. We further leverage their synergy to integrate them together and jointly optimize data rate in presence of these optimizations. Using trace-driven simulation and testbed evaluation, we show that each of these techniques is effective and yields significant throughput gain, and their combination with rate adaptation further increases the gain. The benefit extends to both static and mobile networks.

A number of future directions remain. First, our approach requires CSI information, which is more fine-grained than the widely used SNR and more challenging to predict. To enhance the robustness against prediction errors (e.g., arising from high mobility scenarios), we plan to explore robust optimization (e.g., by applying stochastic optimization and robust traffic engineering techniques developed for the Internet [6]). Second, we are interested in collecting more traces and understanding the benefits of our approaches in more extensive scenarios.

Acknowledgments: This research is supported in part by NSF grants CNS-1017549, CNS-0916106, and CNS-0546755. We are grateful to Daniel Halperin for providing us the tool to measure CSI and helping us to properly use it in our environment. We thank Romit Roy Choudhury and anonymous reviewers for their valuable comments.

8. REFERENCES

- [1] LAN/MAN Standards Committee of the IEEE Computer Society. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. *IEEE Standard 802.11*, 1999.
- [2] LAN/MAN Standards Committee of the IEEE Computer Society. Part 11: Wireless LAN Medium Access Control and Physical Layer (PHY) Specifications. *IEEE Standard 802.11*, 2009. <http://standards.ieee.org/getieee802/download/802.11n-2009.pdf>.
- [3] K. C. Beh, S. Armour, and A. Doufexi. Joint time-frequency domain proportional fair scheduler with HARQ for 3GPP LTE systems. In *Proc. of IEEE VTC*, 2008.
- [4] J. Bicket. Bit-rate selection in wireless networks. In *MIT Master's Thesis*, 2005.
- [5] B. Chen, Z. Zhou, Y. Zhao, and H. Yu. Efficient error estimating coding: Feasibility and applications. In *Proc. of ACM SIGCOMM*, 2010.
- [6] B. Fortz and M. Thorup. Optimizing OSPF/IS-IS weights in a changing world. *IEEE Journal on Selected Areas in Communications (Special Issue on Recent Advances on Fundamentals of Network Management)*, 2002.
- [7] R. Ganti, P. Jayachandran, H. Luo, and T. Abdelzaher. Datalink streaming in wireless sensor networks. In *Proc. of ACM SenSys*, Nov. 2006.
- [8] A. Goldsmith. *Wireless Communications*. Cambridge University Press, 2005.
- [9] D. Halperin, W. Hu, A. Sheth, and D. Wetherall. Predictable 802.11 packet delivery from wireless channel measurements. In *Proc. of ACM SIGCOMM*, 2010.
- [10] G. Holland, N. Vaidya, and P. Bahl. A rate-adaptive MAC protocol for multihop wireless networks. In *Proc. of ACM MobiCom*, Jul. 2001.
- [11] A. L. Intini. Orthogonal frequency division multiplexing for wireless networks standard IEEE 802.11a. *ATON Q1 2001 Progress Reports*, Dec. 2000.
- [12] 802.11n channel measurement tool. <http://ils.intel-research.net/projects/80211n-channel-measurement-tool>.
- [13] A. P. Iyer, G. Deshpande, E. Rozner, A. Bhartia, and L. Qiu. Fast resilient jumbo frames in wireless LANs. In *Proc. of IEEE IWQoS*, Jun. 2009.
- [14] K. Jamieson and H. Balakrishnan. PPR: Partial packet recovery for wireless networks. In *Proc. of ACM SIGCOMM*, 2007.
- [15] P. S. Kalekar. Time series forecasting using Holt-Winters exponential smoothing. Dec. 2004. http://www.it.iitb.ac.in/~praj/acads/seminar/04329008_ExponentialSmoothing.pdf.
- [16] J. Kim, S. Kim, S. Choi, and D. Qiao. CARA: collision-aware rate adaptation for IEEE 802.11 WLANs. In *Proc. of IEEE INFOCOM*, Apr. 2006.
- [17] M. Lacage, M. H. Manshaei, and T. Turletti. IEEE 802.11 rate adaptation: A practical approach. In *Proc. of ACM MSWiM*, Oct. 2004.
- [18] S. B. Lee, I. Pefkianakis, A. Meyerson, S. Xu, and S. Lu. Proportional fair frequency-domain packet scheduling for 3GPP LTE uplink. In *Proc. of IEEE INFOCOM Mini-Conference*, 2009.
- [19] L. E. Li, K. Tan, Y. Xu, H. Viswanathan, and Y. R. Yang. Remap decoding: Simple retransmission permutation can resolve overlapping channel collisions. In *Proc. of ACM MobiCom*, Sept. 2010.
- [20] K. C.-J. Liu, N. Kushman, and D. Katabi. ZipTx: Harnessing partial packets in 802.11 networks. In *Proc. of ACM MobiCom*, 2008.
- [21] A. Miu, H. Balakrishnan, and C. E. Koksal. Improving loss resilience with multi-radio diversity in wireless networks. In *Proc. of ACM MobiCom*, Aug. - Sept. 2005.
- [22] ONOE rate control. http://madwifi.org/browser/trunk/ath_rate/onoe.
- [23] D. Qiao, S. Choi, and K. Shin. Goodput analysis and link adaptation for IEEE 802.11a wireless LANs. *IEEE Transactions on Mobile Computing*, Oct. 2002.
- [24] H. Rahul, F. Edalat, D. Katabi, and C. Sodini. FARA: Frequency-aware rate adaptation and MAC. In *Proc. of ACM MobiCom*, Sept. 2009.
- [25] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly. Opportunistic media access for multirate ad hoc networks. In *Proc. of ACM MobiCom*, Sept. 2002.
- [26] S. Sen, N. Santhapuri, R. R. Choudhury, and S. Nelakuditi. AccuRate: constellation based rate estimation in wireless networks. In *Proc. of NSDI*, 2010.
- [27] S. Sesia, I. Toufik, and M. Baker. *The UMTS Long Term Evolution: From Theory to Practice*. Wiley, 2009.
- [28] T. Starr, J. Cioffi, and P. Silverman. *Understanding Digital Subscriber Line Technology*. Prentice Hall PTR, 1999.
- [29] USRP. <http://www.ettus.com/>.
- [30] M. Vutukuru, H. Balakrishnan, and K. Jamieson. Cross-Layer Wireless Bit Rate Adaptation. In *Proc. of ACM SIGCOMM*, Barcelona, Spain, August 2009.
- [31] S. H. Wong, H. Yang, S. Lu, and V. Bharghavan. Robust rate adaptation in 802.11 wireless networks. In *Proc. of ACM MobiCom*, Sept. 2006.
- [32] G. Woo, P. Kheradpour, D. Shen, and D. Katabi. Beyond the bits: Cooperative packet recovery using physical layer information. In *Proc. of ACM MobiCom*, 2007.
- [33] H. Yang, F. Ren, C. Lin, and J. Zhang. Frequency-domain packet scheduling for 3GPP LTE uplink. In *Proc. of IEEE INFOCOM*, 2010.