Exploring Smart Pilot for Wireless Rate Adaptation

Lu Wang, *Member, IEEE*, Xiaoke Qi, Jiang Xiao, *Member, IEEE*, Kaishun Wu, *Member, IEEE*,
Mounir Hamdi, *Fellow, IEEE*, and Qian Zhang, *Fellow, IEEE*

***Abstract*—Rate adaptation is an essential component in today's wireless standards, which help approach the channel capacity and maximize the throughput. However, how to estimate the optimal data rate in a fluctuated channel remains of great concern. Previous wisdoms leverage PHY layer information for rate estimation, including confidence information like SoftPHY hints, and channel state information (CSI) measurements. However, when experiencing rapid time varying and frequency selective fading channel, the above metrics can be inaccurate. The reason roots from the fact that there are not enough cost-efficient pilots, which are pre-known symbols inserted in a packet for channel estimation. In this paper, we observe that by digging into both PHY layer decoder and upper layer protocol headers, more reliable data bits with high confidence level can be exploited. These data bits, termed smart pilot, can be used to calibrate the channel estimation measurements cost-efficiently. Based on the calibrated estimation, we further propose a novel greedy rate selection algorithm to track the optimal data rate, which successfully avoids the impact of deep fading subcarriers in both legacy 802.11a/g and 802.11n MIMO systems. Our experiments on GNU radio testbed show that SmartPilot quickly tracks the link variance, and improve the channel estimation accuracy by 87%. Furthermore, the trace driven simulation reveals that greedy rate selection algorithm predicts the data rate as good as the optimal rate adaptation algorithms for 802.11 standards.**

***Index Terms*—Rate Adaptation, Channel Estimation, Smart Pilot, Protocol Header.**

I. Introduction

NOWADAYS, wireless local area networks (WLANs) are facing great challenges to meet the increasing user demands for high speed communications. The latest 802.11 standards offer date rates ranging from 6Mbps to 600Mbps through combinations of modulation, coding and spatial streams [1]. Accordingly, rate adaptation protocols are adopted to dynamically adjust the data rate on the basis of the channel quality. However, wireless links often experience a variety of impairments. These impairments add noise, introduce bit errors, or otherwise distort the transmitted signal, and thus make the wireless links extremely unstable both in time domain and frequency domain [2]. This instability brings in a great challenge to rate adaptation design. How to predict such a fluctuated channel and estimate the optimal rate remains an open topic.

The state-of-the-art research primarily focuses on two kinds of metrics for rate adaptation: packet loss rate (PLR) and signal to noise ratio (SNR)- bit error rate (BER). SampleRate [3] and RRAA [4] are two representatives of loss-triggered rate adaptation, which rely on PLR to infer the channel condition. The PLR is calculated by tens or hundreds of frame transmissions, making it not responsive to channel variance. Instead of operating on frame-level, SoftRate [5] utilizes SoftPHY hints to obtain per-packet BER. These hints are the per-bit confidences computed in the decoding decision. Thus, the good bit rates can be chosen on per-packet basis, which is more adaptive to the rapidly time-varying channel. Another short timescale metric is SNR retrieved on each packet reception [6], and is commonly measured by the received signal strength indication (RSSI). Yet RSSI is known to be coarse and insufficient, especially in frequency selective fading channel. To better reflect the channel variation in frequency domain, effective SNR (ESNR) is proposed in [7], which is an estimation of channel state information (CSI). ESNR serves as a promising metric for rate adaptation, since it takes frequency diversity into consideration. However, complete CSI information is costly to obtain and store. Instead of fine-grained CSI, current 802.11n devices support coarse-grained CSI that is computed on a per-packet basis. Thus, there is a certain gap between ESNR and the real CSI.

The origin of failure to track the optimal data rate falls into two aspects. The first one is inaccurate channel estimation, which directly links to the inappropriate rate selection choice. Current rate adaptation protocols are not capable of obtaining the accurate channel status. The fundamental reason lies in the fact that the information they acquire for channel estimation is limited. In today's 802.11 standards, only preamble and a few inserted pilot symbols are utilized for channel estimation in each frame. There always leaves a measurement error that cannot be truly removed. To quantify the channel estimation error, we conduct experiments using a simple USRP2 transmitter-receiver node topology. We compute the CSI based on the received known PN sequences as the real channel staues. Fig. 1 illustrates the estimation error calculated as the Mean Square Error (MSE) between the real channel and the channel

Manuscript received September 6, 2014; revised June 7, 2015 and December 9, 2015; accepted February 27, 2016. Date of publication March 16, 2016; date of current version July 8, 2016. This work was supported in part by grants from China NSFC under Grant 61502313 and Grant 61472259, in part by Shenzhen Science and Technology Foundation under Grant JCYJ20150324141711621 and Grant KQCX20150324160536457, in part by Guangdong Young Talent Project under Grant 2014TQ01X238, in part by the 973 project under Grant 2013CB3296006, in part by the RGC under the contracts CERG MHKUST609/13, 622613, 16212714, and 16203215, and in part by the Qatar National Research Fund under Grant NPRP 6-718-2-298. The associate editor coordinating the review of this paper and approving it for publication was S. Valaee.

L. Wang and K. Wu are with College of Computer Science and Software Engineering, Shenzhen Univeirsty, Shenzhen 518060, China (e-mail: wanglu@szu.edu.cn; wu@szu.edu.cn).

X. Qi is with the Institute of Acoustics, Chinese Academy of Sciences, Beijing, China (e-mail: xiaoke.qi@nlpr.ia.ac.cn).

J. Xiao and Q. Zhang are with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Kowloon 852, Hong Kong (e-mail: jiangxiao@cse.ust.hk; qianzh@cse.ust.hk).

M. Hamdi is with the College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar (e-mail: mhamdi@qf.org.qa).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TWC.2016.2542807

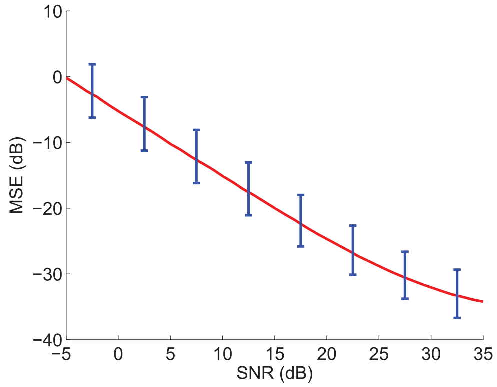


Fig. 1. Channel estimation error using 802.11 preambles, which keeps us away from idea date rate.

measured with 802.11 preamble. This difference keeps us away from tracking the ideal rate. Inserting extra pilots may be a solution, yet it incurs extra control overhead and decreases the effective data rate. We will discuss it later in detail in Sec. III. The second aspect lies in the inability to determine the optimal data rate under frequency diversity. None of the existing metrics, including ESNR and BER, have differentiated the subcarrier impact in legacy 802.11a/g and 802.11n MIMO systems. When some subcarriers experience deep fading, the state-of-the-art will conservatively select a lower data rate to ensure transmissions on all subcarriers. However, the subcarriers without fading lose the opportunity to transmit at a higher data rate that may yield higher throughput. Adaptive modulation and coding with per-subcarrier data rate could solve the problem, yet it complicates the system. Thus researchers still put more efforts on single data rate adaptation.

Motivated by the above observations, we present SmartPilot, a wireless rate adaptation protocol that exploits potential pilots across multiple layers to track the channel variance, and selects the optimal data rate in both legacy 802.11a/g and 802.11n MIMO systems. The SmartPilot design stems from two facts. First, at PHY layer, the decoded bits with relatively high confidence level are regarded reliable. They can be selected as pilots to help alleviate the channel estimation error and approach the real channel status. Second, in 802.11 protocol headers, there are a great number of fields that have fixed values within a certain link [8]. These fields can be harnessed as pilots to further improve the channel estimation. We term the former pilots as Soft Pilots, and the latter ones as Hard Pilots. These smart pilots serve as a built-in strategy to calibrate the channel estimation measurements, say CSI, and better approach the real channel status cost-efficiently. With the calibrated CSI(CCSI), we propose a novel rate selection algorithm termed greedy rate selection (GRS). Unlike the previous methods that weight all the subcarriers as equal for rate selection, GRS differentiates the subcarriers with deep fading, and chooses those with better channel quality to calculate a single optimal data rate. GRS successfully avoids the weaker subcarriers to drag down the data rate to a lower level, and provides more opportunities for stronger subcarriers.

We have implemented SmartPilot protocol in a GNU Radio testbed. Experiments with our software radio prototype verify that SmartPilot can alleviate the channel estimation error by 87% in a cost efficient way. We also conduct trace-driven simulations to evaluate the greedy rate selection over SmartPilot. The results demonstrate that it improves the throughput 1.9× and 1.8× compared with SoftRate and ESNR in 802.11a/g systems, and 1.3× compared with MiRA in 802.11n MIMO systems. The performance gain stems from SmartPilot's ability to quickly track the channel variance, and predict the optimal data rate against multi-path effect. In summary, the main contributions of this paper over the existing rate adaptation protocols are as follows:

- We propose SmartPilot to alleviate the channel estimation error. To the best of our knowledge, this is the first work to exploit decoding data and header bits as pilots to obtain more accurate channel status.
- We present a novel greedy rate selection algorithm based on calibrated channel estimation. The proposed algorithm differentiates the impact of the weaker subcarriers, and leverages more stronger subcarriers for rate selection to maximize the overall throughput.
- We implement SmartPilot on a GNU Radio testbed. Experimental results verify that SmartPilot reduces the channel estimation error by 87%. We also conduct trace-driven simulations to demonstrate the effectiveness of greedy rate selection algorithm in both legacy 802.11a/g and 802.11n MIMO systems.

II. Related Work

The existing wisdoms of rate adaption adjust the data rates according to a certain metric. In SampleRate [3] and RRAA [4], packet loss rate (PLR) is used at the sender side to infer the channel condition. Since FRR is calculated by tens or hundreds of frame transmissions, these two approaches are not so responsive to channel variance. CARA [9] further intergrades 802.11 built-in functionalities to differentiate collisions from rate adaptation malfunction, such as RTS/CTS and Clear Channel Assessment (CCA). Although it is more likely to make rate adaptation decisions, the metric is still irresponsive. On the other hand, SoftRate [5] utilizes SoftPHY hints to obtain per-packet BER, and conducts rate adaptation on packet level. BER is much more adaptive to the rapidly varying channel, yet it has certain drawbacks, e.g., when BER = 0, it is difficult to determine to which level we should increase the data rate. On the otherhand, SmartPilot utilizes calibrated CSI for rate selection to avoid the drawbacks. Also, unlike SoftRate that leverages all the data bits to compute BER, SmartPilot discovers and only leverages the data bits that are proved to be reliable. Thus, it is more accurate and can better track the channel quality.

Another short timescale metric is SNR measured through RSSI on each packet reception. FARA [10] is the first work that puts forward frequency diversity in rate adaptation. It computes per-frequency SNR as the adaptation metric. Even so, SNR obtained from RSSI is known to be coarse and insufficient, especially in frequency selective fading channel. To better reflect the channel variation, ESNR [7] proposes effective SNR computed by CSI for rate adaptation, which takes frequency diversity into consideration. AccuRate [11] also leverages PHY layer information to improve the accuracy of rate selection. Even so, all the previous methods fail to track the optimal data rate, since the information they acquire for rate estimation is not sufficient enough. Also, the rate selection mechanisms do not take frequency diversity into consideration. SmartPilot, which has been initiated in our previous work [12], exploits more pilots to track the link variance with minimum overhead. The subcarrier quality is also differentiated for better rate selection. In this paper, we further investigate SmartPilot under more scenarios, and talk about the critical issues when applying SmartPilot in practise.

More recently, the prevailing of IEEE 802.11n has motivated researchers to develop rate adaptation MIMO systems [13]. As multiplexing and diversity are used in MIMO systems, 802.11n provides a wide range of rate configurations with the increasing number of the antennas. Therefore, rate adaptation becomes more challenging. MiRA [14] is the first work to observe that rate adaptation should exploit the inherent MIMO characteristics. Thus, it proposes an interesting ZigZag search among different MIMO modes to find the rate to optimize throughput. RAMAS [15] categorizes all the rate configurations into two groups, and adapts these two groups concurrently. ARAMIS [16] indicates that the best metric for MIMO rate adaptation is to use 802.11ns Channel State Information (CSI). However, it cannot obtain the accurate CSI, and thus proposes a practical link metric based on RSSI. Unlike the previous methods, SmartPilot has the build-in mechanism to calibrate CSI and obtain more accurate channel estimation measurement, and thus provide more opportunities for MIMO rate adaptation.

III. Channel Estimation Error

Before we jump into the detailed design of SmartPilot, we first introduce the idea of channel estimation error, which is considered as the primary obstacle in rate adaptation. Due to time-varying and frequency selective fading, the transmitted signal is distorted. The existing 802.11 standards estimate the channel response from preamble, and compensate its effect for each subcarrier in one OFDM symbol [7]. Specifically, two identical pseudo-noise (PN) sequences are designated as preamble, each denoted as $\mathbf{x}$ with length N_d. Assuming $\mathbf{y_0^1}$ and $\mathbf{y_0^2}$ are the received sequences of the preamble, and then they can be expressed as,

$$\begin{aligned} y_0^1(p) &= h(p)x(p) + n_0^1(p), \\ y_0^2(p) &= h(p)x(p) + n_0^2(p),\ p = 1, \ldots, N_d, \end{aligned} \tag{1}$$

where n_0^1 and n_0^2 are the Gaussian random variables with variance σ_n^2.

To acquire the channel response $\hat{\mathbf{h}}$, Least square (LS) estimation algorithm is adopted,

$$\hat{h}_0(p) = \frac{y_0^1(p) + y_0^2(p)}{2x(p)} = h(p) + n'(p), \tag{2}$$

where $n'(p) = (n_0^1(p) + n_0^2(p))/2$. Therefore, the variance is calculated as,

$$\sigma_{n'}^2 = E\left\{n'(p)^2\right\} = \frac{1}{4}E\left\{(n_0^1(p) + n_0^2(p))^2\right\} = \frac{\sigma_n^2}{2}. \tag{3}$$

Channel estimation error is the difference between the estimated channel status and the ground truth. We define it as $E\{||\hat{\mathbf{h}} - \mathbf{h}||^2\}$, where $\hat{\mathbf{h}}$ and $\mathbf{h}$ denote the estimated channel response and ground truth channel response, respectively. Therefore, it can be measured as,

$$\begin{aligned} \mathrm{mse}_0 &= E\{||\hat{\mathbf{h}}_\mathbf{0} - \mathbf{h}||^2\} \\ &= \frac{1}{N_d}\sum_{p=1}^{N_d}|\hat{h}(p) - h(p)|^2 = \sigma_n^2/2. \end{aligned} \tag{4}$$

Equation (4) aligns with the simulation results in Fig. 1. When SNR is low, the channel estimation error becomes severe, and amplifies the gap between real channel status and the estimated channel status. Since reliable channel estimation provides appropriate metric for rate selection, minimizing the channel estimation error will definitely improve the performance of rate adaptation protocols.

IV. SmartPilot Design

In this section, we describe the overall architecture of SmartPilot. SmartPilot is compatible with the existing error correcting codes and error recovery schemes. By exploiting as many pilots as possible to approach the real channel status, it aims at selecting the optimal data rate against multipath effect.

A. Overview and Design Challenge

Unlike the previous rate adaptation protocols that treat PSDU (PLCP Service Data Unit) as transparent, SmartPilot investigates the reliable data bits in PSDU as pilots to calibrate the CSI and approach the real channel status. To be specific, SmartPilot exploits two kinds of data bits as pilots. First, from PHY layer decoder, some data bits have relatively high confidence levels. These data bits can be considered reliable and selected as known pilots after decoding. We call them soft pilots since they are extracted using softPHY hints. Second, 802.11 PDSU carries numerous upper layer protocol headers, e.g., MAC header, logical link header, network header, etc. Some fields in these headers often have fixed bit values within a certain link, such as the source and destination MAC addresses, and service types. These bits could be extracted as hard pilots, and leveraged for accurate channel estimation.

The fundamental idea of SmartPilot is: *to exploit as many pilots as possible from transparent data unit, and harness them to estimate the data rate that yields the maximum throughput*. The basic idea seems simple and efficient, yet there remains several challenges for implementation. First, how to utilize SoftPHY hints to choose the proper soft pilots remains a great challenge. As we try to extract as many pilots as possible, there always exists a tradeoff between quantity and quality. Second, unlike soft pilots that naturally spread within a frame across all the subcarriers, hard pilots are mainly located in protocol headers. They may have limited contribution to estimate the entire channel, and need to be carefully spread out within a transmission. Third, after we obtain smart pilots and use them to calibrate the CSI, how to choose the optimal data rate in a frequency selective fading channel remains a great concern.

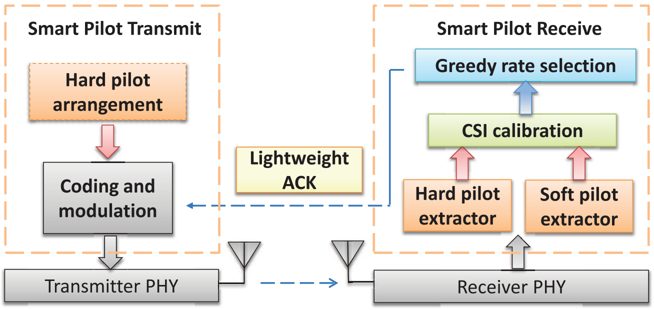
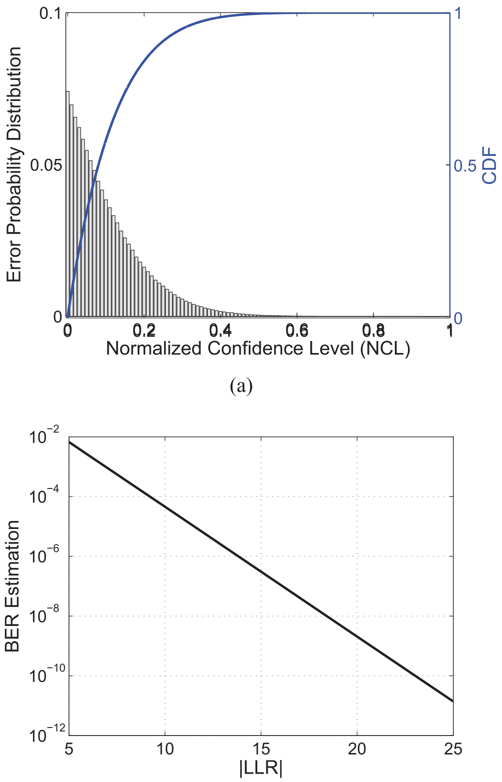


Fig. 2. The block diagrams of SmartPilot sender and receiver. Four important components are included: a) Soft Pilot Extractor, b) Hard Pilot Extractor, c) CSI Calibration, and d) Greedy Rate Selection.

As shown in Fig. 2, SmartPilot has four components to address the above challenges: **Soft Pilot Extractor** that selects soft pilots from PHY layer encoder with guaranteed reliability, **Hard Pilot Extractor** that extracts hard pilots from the upper layer protocol in a cost efficient way, **CSI Calibration** that utilizes smart pilots to approach the real channel status, and **Greedy Rate Selection** that leverages CCSI to compute the highest available date rate in both legacy 802.11a/g and 802.11n MIMO system. The sender first rearranges the messages from upper layer, and then transmits them into air. Upon receiving the data messages, the receiver first extracts hard pilots and soft pilots and store them into the buffer. The pilots are used for CSI calibration, which later feedback to the sender to select the optimal data rate for transmission.



(a)

(b)

Fig. 3. The relationship between bit errors and NCL/LLR. (a) The error probability distribution (b) BER estimation.

B. *Smart Pilot Extraction*

1) Soft Pilot Extraction: Soft pilots are the decoded bits extracted from PHY layer decoder using SoftPHY hints, e.g., posteriori Log-Likelihood Ratios (LLRs) [17]. LLR represents the confidence level of a decoded bit. Intuitively speaking, whenever a bit has high LLR, it has high probability to be correctly decoded. Therefore, LLR gives us an insight to extract decoded bits as soft pilots. LLR can be obtained from the *maximum likelihood* (ML) or *maximum a posteriori probability* (MAP) decoder [18]. We present LLR as L_0 and calculate it as,

$$L_0(i) = \frac{2y_i}{\sigma^2}, i = 1, \ldots, N. \tag{5}$$

where $y_i, i = 1, \ldots, N$ is the output bits from the decoder and σ^2 is the Gaussian noise variance.

One critical problem is how to ensure the reliability of soft pilot extraction using LLRs. Due to channel impairments such as noise variation, channel multipath, and collision, LLRs could be objective and cannot represent the accurate bit confidence level. The normalized confidence level (NCL), defined as $\frac{|LLR|}{\max(|LLR|)}$, is more subjective and can be used as the extraction metric. To verify this point, we conduct simulations to investigate how NCL reflects the error probability of a decoded bit. A sequences of bits are first coded and passed through a fading channel. We then feed them into the decoder.

Fig. 3(a) depicts the relationship between NCL and error probability distribution. The erroneous bits have relatively low NCL, and the bits with higher NCL turn out to be always reliable. To be specific, when NCL is below 0.5, the decoded bits have high probability to be erroneous. After NCL exceeds a certain threshold, e.g., 0.8, the error probability quickly drop to 0, indicating that the decoded bits are correct with no errors. Thus, we use NCL as the soft pilot extract extraction metric. According to SoftRate, the error probability has an approximately exponential decay relationship with $|LLR|$. We demonstrate this relationship in Fig. 3(b). When a decoded bit has $|LLR|$ value higher than 20, its BER is lower than 10^{-9}, which is small enough to be trusted.

We divided the decoding process into several loops. In each loop, decoded bits with $NCL \geq 0.8$ and $|LLR| \geq 20$ are extracted as soft pilots. These soft pilots are also regarded as the known bits in the next loops to improve the decoding performance. During pilot extraction, the pilot values are the hard decisions of the outputs $|LLR|$. However, this method is likely to extract erroneous bits. To avoid this situation, we make the decision based on the decoding status. If the decoding is successful, we choose the hard decisions from current loop as the decoded bits. Otherwise, we choose the values from the previous loops as the decoded bits. That is because the extraction for the first loops is always true, and decoded bits for different loops are true in different positions. So if we average the decoded bits, the errors can be suppressed. Our experiments in Sec. V-A prove its efficiency.

2) Hard Pilot Extraction: Different upper layer protocols have various packet semantics. Since hard pilots exhibit quite distinct features and highly depend on data traffic types, it is not practical for the sender to inform the exact pilot locations to the receiver. To simplify the design and minimize the overhead, we leverage the idea of bit bias to conduct pilot extraction at the receiver side. Bit bias is the high probability that a bit takes value of 0 or 1 [8]. Assume a window of packet $T_{m,n}$ has m packet, each with n bits in a certain link,

$$T_{m,n} = \begin{bmatrix} b_{1,1} & \cdots & b_{1,n} \\ \vdots & \ddots & \vdots \\ b_{m,1} & \cdots & b_{m,n} \end{bmatrix}, \tag{6}$$

the bit bias $\beta_i(T_{m,n})$ of bit i can be computed as:

$$\beta_i(T_{m,n}) = 2 \times \left| \frac{\sum_{k=1}^{m} b_{k,i}}{m} - \frac{1}{2} \right| \tag{7}$$

where m is history size of the received packets, and n is the header length. $b_{i,j}$ defines the bit value of the i^{th} bit in the j^{th} packet. If bit i is fixed to the value 0 or 1 within m packets, $\beta_i(T_{m,n}) = 1$, and then it will be selected as hard pilot and stored in a *hard-pilot* buffer for this link. Otherwise, $\beta_i(T_{m,n}) = 0$. The receiver will keep sniffing the received packets at the running time, and compute the bit bias for each bit located in the first 80 bytes of a packet, which include all protocol headers from MAC layer to the transport layer.

During each transmission, the receiver first updates the *hard-pilot* buffer before using them to calibrate CSI. It is possible that the extracted pilots do not match the actual received bits. To avoid this pilot misprediction and and guarantee the accuracy of the pilots, we also include two CRC4 block checksum for the first and last half the header length. Specifically, the update procedure follows two principles:

1) **Principle 1**: If a block checksum succeeds, the receiver recomputes the bias value $\beta_i(T_{i,n})$ for each bit i.
   a) If $\beta_i(T_{i,n}) = 1$, bit i will be stored in *hard-pilot* buffer.
   b) If $\beta_i(T_{i,n}) < 1$, bit i will be removed from the buffer.
2) **Principle 2**: If a block checksum fails, LLR is used to examine the reliability of each bit.
   a) If the confidence level of bit i exceeds threshold σ, then this bit is considered reliable. The receiver follows principle 1a and 1b for pilot update.
   b) Otherwise, if $|LLR(i)| < \sigma$, bit i will be removed from the buffer if already stored.

As we mentioned before, the hard pilots are extracted from the protocol headers, e.g., within the first 80 bytes of the packets. Therefore, they are not capable of tracking the channel variation of the whole packet. To make them useful, we propose two interleavers at the sender side: *MAC block interleaver* and *PHY symbol interleaver*. MAC interleaver aims to distribute the header bits evenly within the entire Message Protocol Data Unit (MPDU). It divides the header of MPDU into several blocks, and inserts each into a subsequent data block that divided by the encoder (e.g., LDPC encoder). In this way, we can ensure that each coded block has at least one header block after PHY encoding. As illustrated in Fig. 4, the header and data units in MPDU have been divided into three blocks respectively. The row-column interleaver assigns each data block with a header block, and feed them into PHY layer block encoder for further operation. Meanwhile, PHY symbol interleaver is a random interleaver that conducted after symbol mapping. As shown in Fig. 5, it spreads the modulated symbols from header bits to all OFDM symbols uniformly. Since channel estimation is based on symbol-level instead of bit-level, our PHY symbol interleaver distributes the pilot symbols within the entire packet.

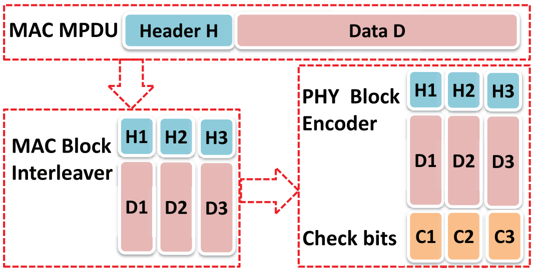


Fig. 4. Illustration of MAC layer block interleaver, from MAC MPDU to PHY layer coded message.

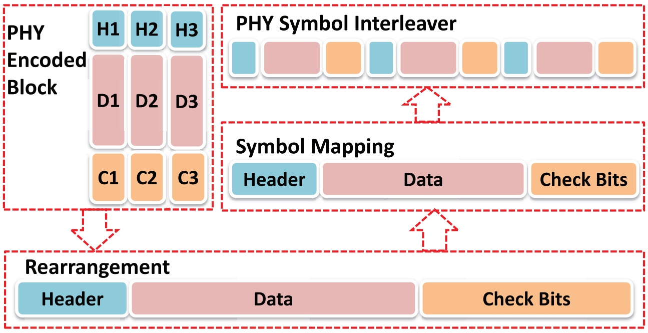


Fig. 5. Illustration of PHY layer symbol interleaver, which operates on the PHY layer coded message.

When a receiver extracts hard pilots from the buffer, it cannot directly used them for channel estimation on the received packet. That is because the bit values of a packet may appear random after encryption and scrambling. Decryption and descrambling the received packet is not practical. To address this issue, upon receiving a packet, the receiver first extracts the hard pilots from the buffer, and then executes the same encryption and scrambling algorithm as the sender. The execution guarantees that the hard pilots directly match the received packet at PHY layer, and ensures that SmartPilot is independent from the underlying encryption and scrambling algorithms. The encryption and scrambling algorithm can be exchanged between sender and receiver in advance.

C. CSI Calibration

CSI calibration utilizes the hard and soft pilots to alleviate the channel estimation error, and thus improves the channel estimation accuracy. Specifically, the estimation obtained from the above pilots is used to calibrate the initial estimation from the preamble. Assume the extracted pilots have a position set

$\boldsymbol{S}_k$ in the kth OFDM symbol. A pilot at the pth subcarrier has transmitted value $x_k(p)$ and received value $y_k(p)$,

$$y_k(p) = h(p)x_k(p) + n_k(p). \tag{8}$$

With the assist of pilots, the LS channel estimation for the k^{th} OFDM symbol is obtained as:

$$\hat{h}_k(p) = \frac{y_k(p)}{x_k(p)} = h(p) + n_k(p),\ p \in \boldsymbol{S}_k,\ k = 1, \ldots, N. \tag{9}$$

The channel estimation is calibrated by weighting the previous and current estimation as,

$$\tilde{h}_k(p) = \begin{cases} \alpha_k(p)\tilde{h}_{k-1}(p) + (1 - \alpha_k(p))\hat{h}_k(p), & \text{for } p \in \boldsymbol{S}_k, \\ \tilde{h}_{k-1}(p), & \text{otherwise.} \end{cases} \tag{10}$$

where $\alpha_k(p)$ is the weight, and $\tilde{h}_0(p) = \hat{h}_0(p)$. Thus, the estimation error after calibration is,

$$\text{mse}_k(p) = E\{||\tilde{h}_k(p) - h(p)||^2\}, \tag{11}$$

and we choose the optimal weight by minimizing the channel estimation error for each subcarrier,

$$\alpha_{k,opt}(p) = \underset{\alpha_k(p)}{\text{argmin}}\ \text{mse}_k(p). \tag{12}$$

To derive the optimal weight, we assume there is S_p^k pilots at the p^{th} subcarrier in k OFDM symbols, and $\boldsymbol{P}$ contains the index of OFDM symbols with S_p^k pilots. With the same noise variance $\sigma_n^2/2$, the minimum mse is obtained when assuming the channel estimation has the same accuracy. Therefore, Equation (9) can be expressed as:

$$\tilde{h}_k(p) = \begin{cases} \beta_k(p)\hat{h}_0(p) + \frac{1-\beta_k(p)}{S_p^k} \sum_{n=1}^{S_p^k} \hat{h}_{\boldsymbol{P}_k}(p), & \text{for } S_p^k > 0 \\ \hat{h}_0(p), & \text{for } S_p^k = 0. \end{cases}$$

where $\beta_k(p)$ is the combined weight, and is expressed as:

$$\begin{aligned} \beta_{k,opt}(p) &= \underset{\beta_k(p)}{\text{argmin}}\ \text{mse}_k(p) \\ &= \underset{\beta_k(p)}{\text{argmin}}\ \beta_k(p)^2\sigma_{n'}^2 + S_p^k\left(\frac{1-\beta_k(p)}{S_p^k}\right)^2\sigma_n^2 \\ &= \underset{\beta_k(p)}{\text{argmin}}\ \sigma_n^2 \cdot \\ &\quad \left(\left(\frac{1}{2} + \frac{1}{S_p^k}\right)\beta_k(p)^2 - \frac{2}{S_p^k}\beta_k(p) + \frac{1}{S_p^k}\right). \end{aligned} \tag{13}$$

Equation (13) infers that the optimal weight can be easily obtained by finding the minimum point of the quadratic function, which is expressed as:

$$\beta_{k,opt}(p) = \frac{2}{S_p^k + 2}. \tag{14}$$

Therefore, we have:

$$\alpha_{k,opt}(p) = 1 - \frac{1 - \beta_{k,opt}(p)}{S_p^k} = \frac{S_p^k + 1}{S_p^k + 2}. \tag{15}$$

And the channel estimation error can be alleviated by:

$$\text{mse}_k(p) = \frac{S_p^k + 1}{S_p^k(S_p^k + 2)}\sigma_n^2 < \sigma_{n'}^2. \tag{16}$$

As the number of pilots increases, CSI becomes more reliable.

Furthermore, to simplify the implementation, an iterative calibrated method is derived. For the $(k-1)$th OFDM symbols, the optimal weighting is easily conducted using (15) as:

$$\alpha_{k-1,opt}(p) = \frac{S_p^{k-1} + 1}{S_p^{k-1} + 2}. \tag{17}$$

If there is a pilot at the pth subcarrier, e.g., $S_p^k = S_p^{k-1} + 1$, then we have:

$$\begin{aligned} \alpha_{k,opt}(p) &= \frac{S_p^k + 1}{S_p^k + 2} = \frac{S_p^{k-1} + 2}{S_p^{k-1} + 3} \\ &= \frac{1}{2 - \alpha_{k-1,opt}(p)}. \end{aligned} \tag{18}$$

Therefore, combing (10) and (18), the iterative channel estimation can be obtained. In this way, we can reduce the channel estimation error and approach the real channel status.

D. Greedy Rate Selection

Greedy Rate Selection (GRS) is an essential component in SmartPilot, because it directly links to the throughput. Previous rate selection algorithms barely consider multipath effect, where some subcarriers experience deep fading, and have rather poor channel quality. When choosing the optimal data rate for transmission, these weaker subcarriers will drag the selected rate to a relatively low level, and ruin the opportunities for those stronger subcarriers to have higher rate for transmission. In this subsection, we first present GRS design for legacy 802.11a/g system. Then we extend the algorithm to see how SmartPilot benefits the 802.11n MIMO system.

1) Rate Adaptation for SISO System: Here we follow the constraint of a single modulation and coding scheme (MCS) per client. The goal is to choose the optimal *averaging data rate* R_{opt} that yields the highest throughput. To achieve this goal, we first define that for subcarrier i, its *affordable data rate* r_i is the highest data rate at which a subcarrier can successfully decode. According to [10], r_i can be easily obtained trough $CCSI_i$ based on the off-line training SNR-bitrate relationship. Therefore, the *averaging data rate* R_i when transmitting with r_i is,

$$R_i = \frac{n_i \times r_i}{N}, \tag{19}$$

where N is the total number of subcarriers, and n_i is the number of subcarriers that have *affordable data rate* greater or equal to r_i.

Our goal is to find out the optimal data rate R_{opt} by maximizing R_i, which is modeled as:

$$\begin{aligned} R_{opt} &= \max_{\{i=1,\ldots,N\}} R_i \\ &= \max_{\{i=1,\ldots,N\}} \frac{n_i \times r_i}{N}. \end{aligned} \tag{20}$$

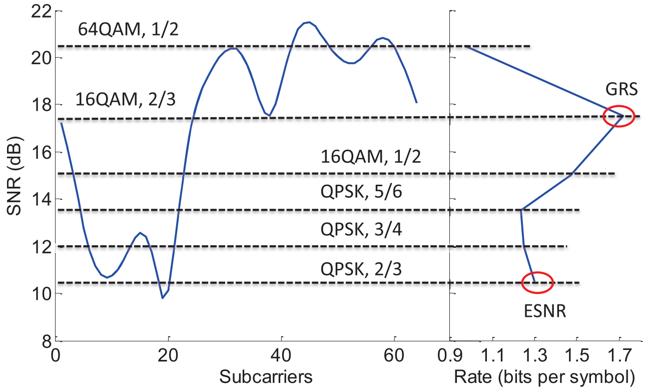


Fig. 6. The motivation of greedy rate selection. The left part depicts the standard SNR-bitrate mapping, and gives an example of a frequency selective fading channel response. The right part is the corresponding averaged data rate under different modulation and coding schemes (MCS).

A brute force method is to compute all the possible R_i and select the largest one. However, it brings in high complexity in time and space domain. To simply the design, we adopt a top-down searching approach. We first classify the subcarriers with the same *affordable data rate* r_i into one groups n_i, and sort all the groups $\boldsymbol{n} = [n_1, n_2, \cdots]$ with $\boldsymbol{r} = [r_1, r_2, \cdots]$ in descending order. Then we start from the group with highest rate r_i, and compute $R_1 = n_1 \times r_1/N$, $R_2 = n_2 \times r_2/N$, etc. We compute all the possible R_i and find out the maximum value. This value will be set to R_{opt}, and will be fed back from the receiver to the sender through acknowledgement.

In a frequency selective fading channel, e.g., some subcarriers experience deep fading, greedy rate selection strategically leaves them out of consideration, and chooses a much higher rate than they can afford. One possible question is how we deal with these deep fading subcarriers. To guarantee their transmission, we assign them the lowest data rate R_{bsc} (e.g., BPSK, 1/2, as preamble). Note that this is simple to achieve. After R_{opt} computation, the receiver will record the subcarriers with $R_i < R_{opt}$, and store their index into a table. This table is fed back along with the rate selection choice R_{opt} to the sender [7]. Upon receiving the feedback, the sender first inquiries the table and modulates the subcarriers with R_{bsc}. And then the rest of the subcarriers are transmitted with R_{opt}. Unlike adaptive modulation schemes that have per subcarrier data rate [10], we only differentiate the deep fading subcarriers, which simplifies the system.

Fig. 6 presents the basic idea of greedy rate selection. The left part denotes the channel quality for 64 subcarriers, while the right part is the averaging data rate under different modulation and coding schemes (MCS). The effective SNR is 11.2 dB, thus ESNR chooses QPSK, 2/3 code rate as its transmission data rate, which is 1.33 bits per symbol. On the contrary, Our GRS chooses the *highest averaging data rate*, which is 16QAM, 2/3 code rate, resulting in a data rate of 1.72 bits per symbol. Furthermore, the subcarriers with *affordable data rate* lower than 16QAM, 2/3 is set to BPSK, 1/2 code rate. Thus the overall data rate is further increased.

2) Rate Adaptation for MIMO System: SmartPilot provides a more accurate estimation of the MIMO channel in a cost efficient way. In this part, we demonstrate how CCSI can inform rate adaptation in MIMO system. The rate selection strategy in MIMO systems is more complex than SISO systems. One primary reason is that all paths undergo different fading. Besides, MIMO system provides many choices apart from adapting different modulation types, such as using spatial multiplexing or transmit diversity, types of guard intervals, and channel width. These configurations exhibits a non-monotonic relationship with data rate. Thus, it is quite challenging to directly map CSI to bitrate.

To benefit from CCSI in the MIMO systems, we incorporate CCSI with other information, such as Packet/Bit Error Rate(PER/BER) [14]. To be specific, CCSI is used to obtain the post-processed SNR (pp-SNR), which is the sub-carrier level SNR after MIMO decoding. Afterwards, we estimate the loss rate (BER) according to the pp-SNR, and use BER as the rate adaptation metric. Essentially this is a BER based rate adaptation algorithm. Yet with CCSI, the estimated BER can be more accurate, and help us better track the ideal data rate. The rate adaptation algorithm includes three steps:

- pp-SNR computation. After CCSI is obtained, the channel on the kth subcarrier from the ith transmitting antenna to jth receiving antenna is estimated as $\tilde{h}_k^{(i,j)}$. Based on the channel estimation, the pp-SNR, defined as the combination of the multiple receptions experienced by the symbol, is calculated for each subcarrier. We adopt a Minimum Mean Squared Error (MMSE) equalizer to compute the processed signal on subcarrier k,

$$\begin{aligned}\tilde{x}_k &= \left(\tilde{H}_k^H\tilde{H}_k + N_0 I\right)^{-1}\tilde{H}_k^H y_k\\ &= \hat{x} + \left(\tilde{H}_k^H\tilde{H}_k + N_0 I\right)^{-1}\tilde{H}_k^H n,\end{aligned} \tag{21}$$

where $\tilde{H}_k$ denotes the channel matrix for the kth subcarrier, whose entry of ith row and jth column is $\tilde{h}_k^{(i,j)}$. Therefore, using a singular value decomposition (SVD), the post-detection noise power is expressed as,

$$\begin{aligned}&E\left[\|\left(\tilde{H}_k^H\tilde{H}_k + N_0 I\right)^{-1}\tilde{H}_k^H n\|^2\right]\\ &= E[\|(V\Sigma^2V^H + N_0 I)^{-1}V\Sigma^2U^Hn\|^2]\\ &= E[\|(V(\Sigma + N_0\Sigma^{-1})^{-1}U^Hn\|^2]\\ &= E[tr((V(\Sigma + N_0\Sigma^{-1})^{-1}U^Hnn^HU(\Sigma + N_0\Sigma^{-1}))]\\ &= N_0 \cdot tr\left((\Sigma + N_0\Sigma^{-1})^{-2}\right).\end{aligned} \tag{22}$$

where $\tilde{H}_k = U^H\Sigma V$, Σ is a rectangular matrix, whose diagonal elements are the singular values of the matrix $\tilde{H}_k$. $tr(\cdot)$ means the trace operation. As a result, the pp-SNR on subcarrier k is calculated as,

$$SNR_k = \frac{E_k}{N_tN_0 \cdot tr\left((\Sigma + N_0\Sigma^{-1})^{-2}\right)}, \tag{23}$$

where N_t is the number of transmit antennas. E_k is the total transmission energy across N_t transmit antennas on subcarrier k, and N_0 is the noise power.

- Error rate estimation. In this step, we obtain the coded loss rate from pp-SNR. To be specific, the pp-SNR is mapped to the uncoded BER using the well-known relationship between SNR and BER [19]. The coded BER in LDPC decoder has not been properly solved. As a design guide, we derive the coded BER in Viterbi decoder with hard decision. Since the performance of LDPC decoder is much better than Viterbi decoder, the derived coded BER serve as an error-probability upper bound [20]. We define p as the uncoded BER, and P_d as the probability of selecting a code word what is Hamming distance d from the correct word. Then $BER_{coded}(p)$ is,

$$BER_{coded}(p) = \sum_{d=d_{free}}^{\infty} a_d P_d(p) \tag{24}$$

When d is even,

$$P_d(p) = \sum_{i=\frac{d+1}{2}}^{d} \binom{d}{i} p^i (1-p)^{d-i} \tag{25}$$

When d is odd,

$$P_d(p) = \frac{1}{2}\binom{d}{\frac{d}{2}} p^{\frac{d}{2}}(1-p)^{\frac{d}{2}} + \sum_{i=\frac{d+1}{2}}^{d} \binom{d}{i} p^i (1-p)^{d-i} \tag{26}$$

- Rate adaptation. Finally, we choose the optimal rate based on the estimated coded BER. The rate selection algorithm follows the basic rule of our greedy rate selection in SISO system. Specifically, for MIMO system with multiple streams, all the possible transmission modes and MCSs are measured based on the BER. The greedy rate selection algorithm zigzags between intra- and inter-mode rate options. Finally we choose a MCS that yields the maximum delivery rate. For the weaker subcarriers that cannot afford the chosen configuration, e.g., the chosen rate is higher than its affordable rate, we simply use BPSK 1/2 to ensure the basic transmission.

V. Performance Evaluation

In this section, we evaluate the performance of SmartPilot through extensive experiments and simulations. Our 802.11a/g/n like PHY layer is built on top of OFDM modules on GNU radio platform [21]. The universal software radio peripheral 2 (USRP2) uses the RFX2450 daughterboard as RF frontend, which operates in the 2.4-2.5GHZ range. Different sets of USRP2 nodes were tested to verify the experiment. Our idea of utilizing soft and hard pilots to calibrate CSI is implemented on the GNU/USRP2 testbed by replaying real-life packet traces in our office. Yet, the high latency incurred in procuring RF samples from the USRP front-end makes it impractical to evaluate the greedy rate selection algorithm in real time [11]. Therefore, we conduct trace-driven simulations by replaying SIGCOMM'08 trace [22] on our interconnected simulator with C++ and Matlab, where the traces were collected in WLANs of hundreds of users attending the conferences. SoftRate [5], ESNR [7] and MiRA [14] are selected as comparisons. Their simulations are also based on the traces we have collected. Finally, we compare SmartPilot with 802.11 standard in terms of optimal delivered rate.

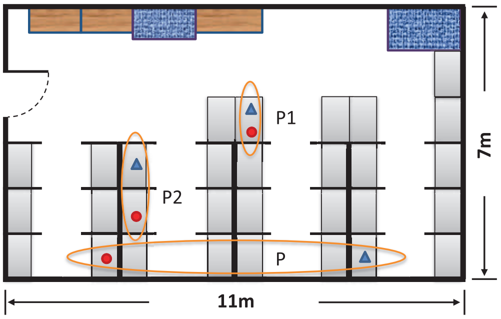


Fig. 7. Experimental environment. 3 sets of nodes are distributed in different locations.)

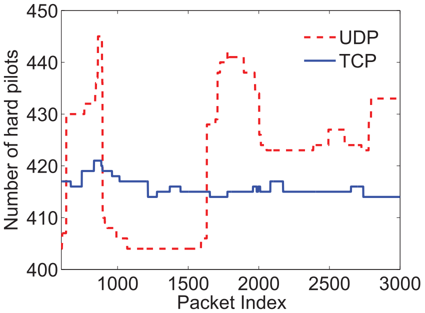


Fig. 8. The number of extracted hard pilot in the first 60 bytes from 3000 packets under different traffic.

A. Pilot Extractor

To begin with, we evaluate the performance of pilot extractor on our GNU Radio platform. Pilot extractor consists of two components: hard pilot and soft pilot extraction. We replayed the SIGCOMM'08 trace in our office, and collected the UDP and TCP traffic for evaluation.

The hard pilots are extracted from the first 60 bytes of packets, which comprise of protocol headers up to transport layer. Fig. 8 shows the number of hard pilots in the first 60 bytes in 3000 packets. The history size is set to 600, with the misprediction threshold of 0.005. The hard pilot remains a constantly high quantity throughout the entire transmission. Specifically, the UDP traffic has an average number of 420-bit hard pilots, which is approximate 87% of the header bits (480 bits). In the meanwhile, the TCP traffic has bit less number of hard pilots. That is because TCP header has more control fields that changed on packet level. However, TCP's hard pilot still yields about 410-bit and 85% of the entire header bits. Therefore, hard pilots from protocol headers are great help for CSI calibration.

When the misprediction rate exceeds 0.05, the extracted hard pilots are not reliable. Specifically, the false positive ones provide opposite pilot information, and thus degrades the channel

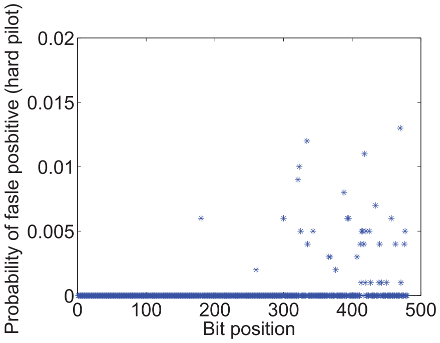


Fig. 9. False positive rate of hard pilots under multiple transmissions.

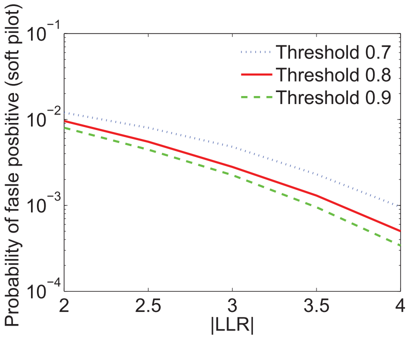


Fig. 10. The number of extracted soft pilot under different LLR thresholds.

estimation accuracy, especially when there are multiple incoming packets from different senders. Fig. 9 plots the false positive rate among the first 60bytes of the entire header. The results are measured using 10000 packets collected on a WLAN downlink. Only 11 hard pilots have false positive rate higher than 0.05. Compared with the total number of 440 hard pilots, the false positive rate is negligible, which validates that with link addressing and a proper extraction history size, the false positive rate can be reduced to less than 0.05, and thus ensure the reliability of hard pilot extraction.

On the other hand, soft pilots are extracted from decoded bits with high LLR values. The threshold has certain impact on the resulting soft pilot. We demonstrate the false alarm rate of extracted soft pilots in Fig. 10, as a function of LLR thresholds. Not surprisingly, higher LLR threshold filters out more unreliable bits, and results in a lower false alarm probability, yet in the meanwhile it produces less soft pilots. To balance the tradeoff between quantity and quality, we take 0.8 as a desirable threshold.

B. Channel Estimation Error

After obtaining the smart pilots, we see how they work on CSI calibration. In order to set a baseline, we trained the link between two USRP nodes by transmitting known 50 long PN sequences. The training sequences are used to compute the CSI as the ground truth. Then we conduct normal data transmission to extract smart pilots. We use least-square estimation (LS) channel estimation algorithm to compute the original CSI, and apply smart pilot for calibration .

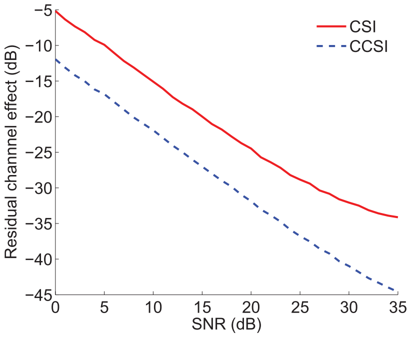


Fig. 11. The channel estimation error between CSI - ground truth and CCSI - ground truth.

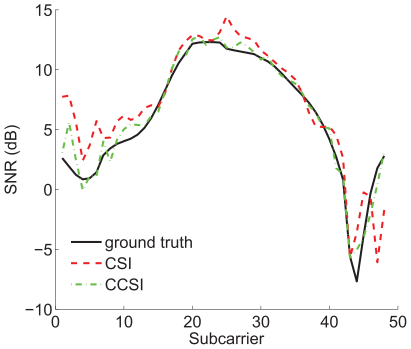


Fig. 12. The subcarrier SNR comparison of ground truth, CSI and CCSI. BPSK modulation, SNR=10 dB.

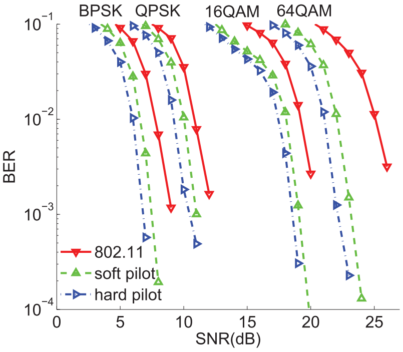
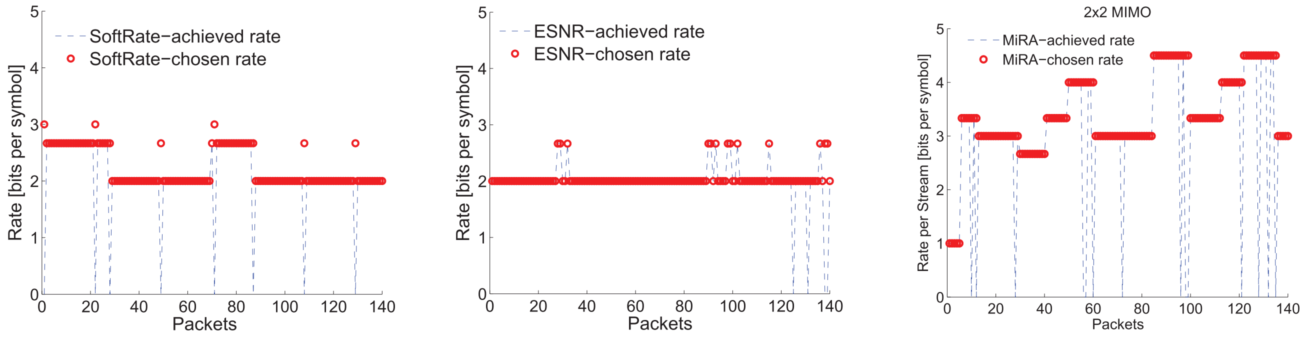


Fig. 13. The BER comparison between the standard 802.11, soft pilot and hard pilot, 3/4 LDPC code rate.

Fig. 11 plots the channel estimation error of CSI and CCSI. CCSI achieves at least 7 dB improvement over CSI across the SNR range from 0 dB up to 35 dB. Furthermore, with higher SNRs, CSI suffers from more critical noise enhancement. That is because noise at the deep fading channel conditions has more influence on channel estimation. On the contrary, calibrated channel estimation is not affected by such influence. It verifies that with the assistance of smart pilots, CCSI is more reliable for rate adaptation.

Fig. 12 presents a comparison among the channel response of ground truth, CSI and CCSI. We use BPSK modulation, and the SNR is around 10 dB. Compared with CSI, CCSI is more approaching to the ground truth. These results verify that SmartPilot can effectively reduce the channel estimation error, and approach the real channel statues.



(a) (b) (c)

Fig. 14. Rate selection comparison under time-varying channel. (a) SoftRate (b) ESNR (c) MiRA.

C. Greedy Rate Selection

Due to the latency constraint of USRP2, we are not allowed to conduct the real time evaluation for rate adaptation protocols. Therefore, we conduct trace-driven simulations to evaluate the performance of smart pilot. We replayed the SIGCOMM'08 trace in our office, and collected all the data traffic for simulation. SampleRate SoftRate and ESNR are chosen as three comparisons.

1) BER Performance: Besides rate estimation, smart pilots can also be leveraged to improve the decoding performance. Thus we first compare the BER performance among the standard 802.11, soft pilot and hard pilot. All available modulation schemes with 3/4 LDPC code rate are implemented. As shown in Fig. 13, hard pilot method has the best BER performance. It can correct bit errors even in deep fading. On the other hand, since only the reliable LLRs are utilized for decoding, the performance of soft pilot has about 0.5 dB less than hard pilot. Moreover, the hard pilot and soft pilot methods are about 1.7 dB and 1.2 dB better than the other one, respectively. Furthermore, more improvement is observed for 64QAM modulation, which has about 4.3 dB and 3.3 dB SNR gains for hard pilot and soft pilot. At the SNR of 8 dB and BPSK modulation, soft pilot and hard pilot respectively decrease 85% and 98% errors. It infers that SmartPilot is also powerful to reduce BER.

2) SmartPilot Rate Adaptation: In this step, we compare the performance of rate adaptation among SmartPilot, SoftRate, ESNR and MIRA over the traces we gathered through USRP. As shown in Fig. 14, the chosen rate is the one computed through rate adaptation metric, where is the estimated BER in SoftRate, effective SNR in ESNR and the Sub-Frame Error Rate in MiRA. As for SmartPilot, two metrics are evaluated, ground truth information and calibrated CSI. The achieved rate is the average data rate for all the packets, which is computed from the transmitted error-free packets. Whenever $BER > 0$, we set the achieved rate to 0.

We compare the performance of SmartPilot with SoftRate, ESNR and MiRA. As shown in Fig. 14, the chosen rates are not satisfactory in all the above three protocols. ESNR outperforms SoftRate in SISO systems, which traces the chosen rate pretty well in fast time varying channel with a rate of 2 bits per symbol. Yet it still cannot catch up with SmartPilot since it cannot reduce the channel estimation error. As for SoftRate, it exhibits very irresponsive reaction, and only achieves a rate of 1.9 bits per symbol due to deep fading subcarriers. In MIMO systems, MiRA achieves a rate of 3.1 bits per symbol, indicating that its zigzag rate probing exploring the inherent MIMO characteristics among different modes. However, the upward rate probing makes is quite unstable to converge an optimal rate. We dig into the detail of these protocols and find out that, the reason for their poor behavior results from the incorrect channel estimation. They are likely to generate the estimated error and thus move on to an incorrect rate.

3) Optimal Data Rate: Finally, we evaluate the performance of SmartPilot in terms of optimal data rate compared the legacy 802.11a/g SISO systems and 802.11n MIMO systems. The optimal data rate is the envelope of all 802.11 available rates. It can be easily obtained by measuring with a fixed configuration.

We first evaluate the performance of SmartPilot over a approximate AWGN channel. This is obtained by replying the traces using transmitter-receiver pair P1 in Fig. 7. The close distance ensures that white noise follows Gaussian distribution, and the energy spreads across the entire channel. The transmitter transmits 10min packets for every rate. The receiver calculates the achieved data rate of each available modulation and code type, and choose the envelope as the optimal data rate. We gauge the achieved data rate of SmartPilot and legacy 802.11a/g SISO system in Fig. 15(a). It is seen that SmartPilot has a better performance than the legacy 802.11a/g when taking advantages of pilots, and obtains a much higher rate over the entire SNR range. It outperforms 802.11 by 56% (-5 dB to 5 dB), 25% (5 dB to 15 dB) and 5% (above 15 dB) achieved data rate. These gains are benefited from hard pilot and soft pilot in LDPC decoder since there is no channel effect over AWGN channel.

Fig. 15(b) demonstrates a performance comparison between SmartPilot and legacy 802.11a/g SISO systems over frequency selective channels. This is obtained by replying the traces using transmitter-receiver pair P2 and P3 in Fig. 7. The results shows that SmartPilot greatly outperforms legacy 802.11 and reaches error-free transmission even at lower SNRs, e.g., 3 dB for BPSK modulation and 1/2 code rate. At the lower SNRs, i.e., less than 5 dB, SmartPilot can achieve data rate 39.5 times than 802.11. That is because the rate is unavailable for 802.11 at the low SNRs, while SmartPilot can successfully recover the message

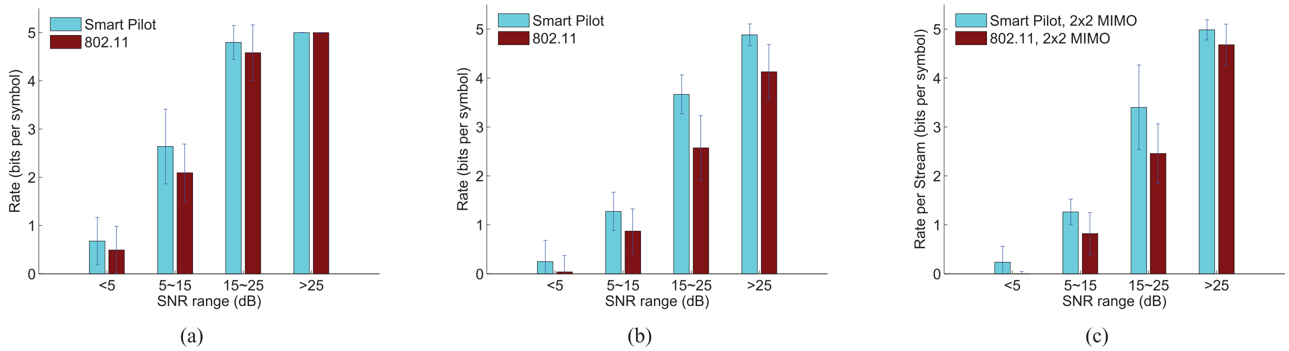


Fig. 15. Rate achieved by SmartPilot and the standard 802.11 standards (a) SISO systems in approximate AWGN channel (b) SISO systems in frequency selective channel (c) MIMO systems in frequency selective channel.

with the aid of pilots, and thus achieves 1.7×, 1.3× and 1.06× performance gains over 802.11 at the range of 5 to 15 dB, 15 to 25 dB and above 25 dB, are obtained by SmartPilot.

Finally, we compare the performance of SmartPilot and 802.11n MIMO systems over frequency selective channels. The traces are also obtained by replying the traces using transmitter-receiver pair P2 and P3 in Fig. 7. We compute the rate achieved per stream as the comparison metric. The results in Fig. 15(c) shows that SmartPilot still outperforms 802.11n under all SNR ranges. To be specific, SmartPilot performs much better under low SNR ranges, e.g, below 15 dB. The performance gain over 802.11n indicates that with more accurate channel estimation, the rate adaptation algorithm can achieve a better performance.

VI. Conclusion

In this paper, we propose a novel rate adaptation protocol termed SmartPilot. It aims at exploiting the potential data bits in PHY layer decoder and upper layer protocol header for rate adaptation. We first identify the problem in the existing rate adaptation protocols, that is, the lack of sufficient information to alleviate the channel estimation error. By investigating soft and hard pilots for channel estimation, SmartPilot can calibrate the channel state information (CSI) in a cost-efficient way. With calibrated CSI, a Greedy Rate Selection algorithm is proposed, which leverages frequency diversity to obtain the optimal data rate for both legacy 802.11a/g and 802.11n MIMO systems. We have verified the efficiency of SmartPilot using GNU/USRP2 platform. The experiment results show that the channel estimation error has been reduced 87% using SmartPilot. We also conduct trace-driven simulations for greedy rate selection algorithm based on CCSI using interconnected Matlab and C++. Extensive results show that SmartPilot achieves 1.9×, 1.8× and 1.3× throughput gain over SoftRate, ESNR and MiRA respectively.

The design of SmartPilot provides a new panel to obtain reliable control information across several protocol layers. In next stage, we propose to exploit smart pilots to benefit more communication systems [23], [24].

References

[1] I. W. Group *et al.* IEEE 802.11n-2009: Enhancements for Higher Throughput, 2009.

[2] C. Wong, R. Cheng, K. Lataief, and R. Murch, "Multiuser OFDM with adaptive subcarrier, bit, and power allocation," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 10, pp. 1747–1758, Oct. 1999.

[3] J. C. Bicket, "Bit-rate selection in wireless networks," Ph.D. dissertation, Dept. Electr. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, 2005.

[4] S. Wong, H. Yang, S. Lu, and V. Bharghavan, "Robust rate adaptation for 802.11 wireless networks," in *Proc. ACM Mobicom*, 2006, pp. 146–157.

[5] M. Vutukuru, H. Balakrishnan, and K. Jamieson, "Cross-layer wireless bit rate adaptation," in *Proc. ACM SIGCOMM*, 2009, pp. 3–14.

[6] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly, "Opportunistic media access for multirate ad hoc networks," in *Proc. 8th Annu. Int. Conf. Mobile Comput. Netw.*, 2002, pp. 24–35.

[7] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Predictable 802.11 packet delivery from wireless channel measurements," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 4, pp. 159–170, 2010.

[8] J. Huang, Y. Wang, and G. Xing, "Lead: Leveraging protocol signatures for improving wireless link performance," in *Proc. 11th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2013, pp. 333–346.

[9] J. Kim, S. Kim, S. Choi, and D. Qiao, "CARA: Collision-aware rate adaptation for IEEE 802.11 WLANs," in *Proc. INFOCOM*, 2006, vol. 6, pp. 1–11.

[10] H. Rahul, F. Edalat, D. Katabi, and C. G. Sodini, "Frequency-aware rate adaptation and MAC protocols," in *Proc. 15th Annu. Int. Conf. Mobile Comput. Netw.*, 2009, pp. 193–204.

[11] S. Sen, N. Santhapuri, R. R. Choudhury, and S. Nelakuditi, "Accurate: Constellation based rate estimation in wireless networks," in *Proc. Conf. Netw. Syst. Des. Implement. (NSDI)*, 2010, pp. 175–190.

[12] L. Wang, X. Qi, J. Xiao, K. Wu, M. Hamdi, and Q. Zhang, "Wireless rate adaptation via smart pilot," in *Proc. IEEE 22nd Int. Conf. Netw. Protocols (ICNP)*, 2014, pp. 409–420.

[13] G. Wang, S. Zhang, K. Wu, Q. Zhang, and L. M. Ni, "TiM: Fine-grained rate adaptation in WLANs," in *Proc. IEEE 34th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2014, pp. 577–586.

[14] I. Pefkianakis, Y. Hu, S. H. Wong, H. Yang, and S. Lu, "MIMO rate adaptation in 802.11 n wireless networks," in *Proc. 16th Annu. Int. Conf. Mobile Comput. Netw.*, 2010, pp. 257–268.

[15] D. Nguyen and J. Garcia-Luna-Aceves, "A practical approach to rate adaptation for multi-antenna systems," in *Proc. 19th IEEE Int. Conf. Netw. Protocols (ICNP)*, 2011, pp. 331–340.

[16] L. Deek, E. Garcia-Villegas, E. Belding, S.-J. Lee, and K. Almeroth, "Joint rate and channel width adaptation for 802.11 MIMO wireless networks," in *Proc. 10th Annu. IEEE Commun. Soc. Conf. Sensor Mesh Ad Hoc Commun. Netw. (SECON)*, 2013, pp. 167–175.

[17] G. D. Forney, Jr., "The Viterbi algorithm," *Proc. IEEE*, vol. 61, no. 3, pp. 268–278, Mar. 1973.

[18] J. Hagenauer and P. A. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," in *Proc. IEEE GLOBECOM*, 1989, pp. 1680–1686.

[19] J. G. Proakis, *Digital Communications*, 5th ed. New York, NY, USA: McGraw-Hill, 2007.

[20] D. Qiao, S. Choi, and K. G. Shin, "Goodput analysis and link adaptation for IEEE 802.11 a wireless lans," *IEEE Trans. Mobile Comput.*, vol. 1, no. 4, pp. 278–292, Oct./Dec. 2002.

[21] E. Blossom, "GNU radio: Tools for exploring the radio frequency spectrum," *Linux J.*, vol. 2004, no. 122, p. 4, 2004.

[22] A. Schulman, D. Levin, and N. Spring, "CRAWDAD data set UMD/SIGCOMM2008 (v. 2009-03-02)," 2009.

[23] L. Wang, K. Wu, J. Xiao, and M. Hamdi, "Harnessing frequency domain for cooperative sensing and multi-channel contention in CRAHNs," *IEEE Trans. Wireless Commun.*, vol. 13, no. 1, pp. 440–449, Jan. 2014.

[24] X. Ji, J. Wang, M. Liu, Y. Yan, P. Yang, and Y. Liu, "Hitchhike: A preamble-based control plane for SNR-sensitive wireless networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 2, pp. 1239–1251, Feb. 2016.

Lu Wang (M'11) received the B.S. degree in communication engineering from Nankai University, Tianjin, China, in 2009, and the Ph.D. degree in computer science and engineering from Hong Kong University of Science and Technology, Hong Kong, in 2013. She is currently an Assistant Professor with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. Her research interests include wireless communications and mobile computing.

Xiaoke Qi received the B.S. degree in communication engineering from Nankai University, Tianjin, China, in 2009, and the Ph.D. degree in signal and information processing from Marine Information Technology Laboratory, Institute of Acoustics, Chinese Academy of Sciences, Beijing, China. She is currently an Assistant Professor with National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. Her research interests include adaptive equalization, rateless coding, rate adaptation, cross-layer optimation, speech assessment, and spatial hearing.

Jiang Xiao (M'11) received the Ph.D. degree in computer science and engineering from Hong Kong University of Science and Technology, Hong Kong, in 2013. She is currently a Research Assistant Professor with HKUST Fok Ying Tung Graduate School, Hong Kong. Her research interests include mobile computing and big data.

Kaishun Wu (M'11) received the Ph.D. degree in computer science and engineering from Hong Kong University of Science and Technology, Hong Kong, in 2011. After that, he worked as a Research Assistant Professor with the Hong Kong University of Science and Technology, Hong Kong. In 2013, he joined Shenzhen University, Shenzhen, China, as a Distinguished Professor. He has co-authored 2 books and published 80 refereed papers in international leading journals and primer conferences. He is the inventor of 6 U.S. and 43 Chinese pending patents (13 are issued). He was the recipient of the Best Paper Awards at IEEE Globecom 2012, IEEE ICPADS 2012, and IEEE MASS 2014, and the 2014 IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award and also selected as 1000 Talent Plan for Young Researchers.

Mounir Hamdi (F'11) received the B.S. degree in electrical engineering–computer engineering minor (with distinction) from the University of Louisiana, Lafayette, LA, USA, in 1985, and the M.S. and Ph.D. degrees in electrical engineering from the University of Pittsburgh, Pittsburgh, PA, USA, in 1987 and 1991, respectively. He is the Founding Dean of the College of Science and Engineering, Hamad Bin Khalifa University (HBKU), Doha, Qatar. Before joining HBKU, he was Chair Professor at the Hong Kong University of Science and Technology (HKUST), and the Head of the Department of Computer Science and Engineering. He is/was on the Editorial Board of various prestigious journals and magazines including the IEEE TRANSACTIONS ON COMMUNICATIONS, the *IEEE Communication Magazine*, *Computer Networks*, *Wireless Communications and Mobile Computing*, and *Parallel Computing*. He has chaired more than 20 international conferences and workshops, and has been on the program committees of more than 200 international conferences and workshops. He was the recipient of the Best Paper Award at the IEEE International Conference on Communications in 2009, IEEE Globecom in 2011, and the IEEE International Conference on Information and Networking in 1998.

Qian Zhang (F'12) received the B.S., M.S., and Ph.D. degrees from Wuhan University, Wuhan, China, in 1994, 1996, and 1999, respectively, all in computer science. He joined Hong Kong University of Science and Technology in September 2005, where she is a Full Professor with the Department of Computer Science and Engineering. Before that, she was with Microsoft Research Asia, Beijing, China, from July 1999, where she was the Research Manager of the Wireless and Networking Group. She has authored about 300 refereed papers in international leading journals and key conferences in the areas of wireless/Internet multimedia networking, wireless communications and networking, wireless sensor networks, and overlay networking. Her research interests include cognitive and co-operative networks, dynamic spectrum access and management, as well as wireless sensor networks. She was the recipient of MIT TR100 (MIT Technology Review) World's Top Young Innovator Award, and the Best Asia Pacific (AP) Young Researcher Award elected by IEEE Communication Society in 2004.