TiM: Fine-Grained Rate Adaptation in WLANs

Guanhua Wang*, Shanfeng Zhang*, Kaishun Wu*[†‡], Qian Zhang* and Lionel M. Ni*[§]

*Department of Computer Science and Engineering
[§]Guangzhou HKUST Fok Ying Tung Research Institute
Hong Kong University of Science and Technology
[†]College of Computer Science and Software Engineering, Shenzhen University
[‡]Corresponding Author
{gwangab, szhangai, kwinson, qianzh, ni}@cse.ust.hk

Abstract—Channel condition varies frequently in wireless networks. To achieve good performance, devices need rate adaptation. In rate adaptation, choosing proper modulation schemes based on channel conditions is vital to the transmission performance. However, due to the natural character of discrete modulation types and continuous varied link conditions, we cannot make a one-to-one mapping from modulation schemes to channel conditions. This matching gap causes either over-select or under-select modulation schemes which limits throughput performance. To fill-in the gap, we propose TiM (Time-line Modulation), a novel 3-Dimensional modulation scheme by adding time dimension into current amplitude-phase domain schemes. With estimation of channel condition, TiM changes base-band data transmission time by artificially interpolating values between original data points without changing amplitude-phase domain modulation type. We implemented TiM on USRP2 and conducted comprehensive simulations. Results show that, compared with rate adaptation choosing from traditional modulation schemes, TiM can improve channel utilization up to 200%.

Keywords—*Adapting Interpolation Rate, Modulation Scheme, Rate Adaptation*

I. INTRODUCTION

Wireless communication suffers from continuously varied link condition, which leads to packet loss or bit errors. This time-varying problem is the main issue that limits wireless link's performance. The movement of link nodes and background interference make this time-varying issue even worse. To achieve good channel utilization in such severe conditions, sender needs to select the highest transmission rate that current channel can support, and dynamically adapt the rate to meet with the continuously varied link condition. This procedure is called rate adaptation. Given its widely deployment in WLANs (Wireless Local Area Networks) and mesh networks, rate adaptation plays a vital role in wireless networks.

A large quantity of recent researches advance rate adaptation. Most of them focus on channel condition estimation, like Signal-to-Noise Ratio (SNR) [24] [10] [18], Bit Error Rate (BER) [17] [12]. Based on the channel condition they estimate, they pick up the corresponding modulation scheme for proper transmission rate. Recent literature can estimate channel condition with high accuracy and to some extent, make relatively full use of existed modulation schemes. However, none of them ever focuses on whether existed modulation schemes we choose from are good enough. So here raises a natural question, "can we push the limit of rate adaptation by
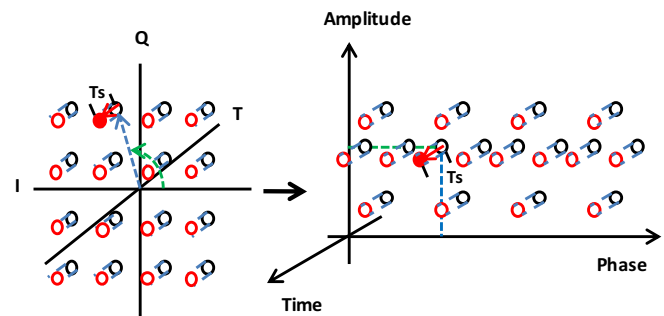

Fig. 1: An example of modified 16-QAM modulation scheme with TiM's additional time dimension.

modifying modulation schemes we choose from and get more throughput gain?"

Properly choosing modulation schemes is a crucial issue in rate adaptation. This choosing process can be regarded as a procedure of mapping different modulation schemes to varied channel conditions. Since the value (e.g. SNR) represents channel condition is continuous whereas modulation types are discrete, we cannot make a perfect one-to-one matching from modulation schemes to channel conditions. Because of this non-perfect matching, we may under-select or over-select a scheme that cannot use the bandwidth efficiently. We evaluate the residual between the scheme can utilize and the current channel condition can really support. This link margin is significantly large. Given this modulation gap, we can revise current modulation schemes to move a step forward of rate adaptation and get better channel utilization.

In this paper, we propose a novel modulation scheme Time-line Modulation (TiM), to fix with the non-perfect matching problem. The widely used modulation schemes in WLANs are designed mainly in amplitude-phase domain, such as BPSK (Binary Phase Shift Keying), QPSK (Quadrature Phase Shift Keying), 16-QAM (Quadrature Amplitude Modulation) and 64-QAM. Based on this 2 Dimension (2D) scheme, TiM adds time dimension to make it into 3D. As depicted in Fig.1, in the traditional 16-QAM modulation scheme, it only has shift in amplitude and phase. The green line refers to the phase shift value whereas the blue line stands for amplitude value.

TiM adds time domain as the transmission time of one data symbol, namely Ts, which is shown in Fig.1 as the red arrow. This 2D to 3D expansion can allow us to insert modulation schemes between two existed adjacent schemes (e.g. QPSK and 16-QAM) on time domain without changing anything on traditional amplitude-phase domain.

Some articles also propose to leverage time-domain diversity in modulation [23] [7]. All of them are using time domain diversity to convey information. More precisely, they leverage different time intervals between transmitted symbols to represent information. However, this kind of time-domain leverage indeed cannot improve signal's resistance to noise and interference. In TiM, the length value in time-domain (i.e. Ts) can be adjusted by adapting the quantity of interpolation values that are inserted between real data points. And the interpolation values can enhance emitted signal to be more robust to interference and more likely to be correctly recovered on the receiver side.

TiM can adjust fine-grained level by regulating the quantity of interpolating values between two adjacent sampling data points. However, how to estimate the fine-grained level is a critical issue. Since with higher fine-grained level, rate adaptation process may change modulation schemes more frequently and the changing overhead will be higher. On the other hand, more fine-grained level means more perfect matching of modulation schemes with channel conditions which leads to more throughput gain. Here we design a *Grain Size Estimation* scheme to make the trade-off between the overhead of changing rate and fine-grained modulation gain.

Since changing modulation schemes on TiM's time domain will have time delay, how to manage this modulation changing overhead is another problem. TiM allows rate changing more efficiently by integrating a simple but useful *Lengthen Coordinator* module. It leverages processing delay and rate changing delay to cancel each other and reduce the overhead.

One point needs to be claimed is that adding time-dimension doesn't mean TiM can achieve real continuous rate adaptation. This is because of two reasons. First, we cannot use a fixed number of bits to represent continuous values which are infinite in system design. Second, there is no need to achieve this which has high overhead without getting significant gain. TiM indeed is also a discrete modulation scheme.

Another point worth mentioning is that TiM indeed does not change the duration time of each symbol. With one specific bitrate, by inserting interpolation points between real data points during transmission, it is as if each real data point's transmission time is lengthened. Thus TiM will not cause spectrum width changing, which is totally different from the bandwidth varying schemes such as SampleWidth [6].

We have implemented TiM scheme on USRP2 platform [2]. We also conduct extensive simulations on Matlab for analysing TiM's performance. The results show that, without changing current 2D modulation schemes, by adding time domain, TiM can achieve up to 200% goodput compared with traditional schemes. For real world wireless traffic, TiM can achieve channel utilization efficiency to nearly 160% on average. Furthermore, *Grain Size Estimation* and *Lengthen Coordinator*'s assistance enhances TiM's performance. To sum up, this paper's contributions are mainly as follows.

- To the best of our knowledge, TiM is the first to propose 3D modulation scheme that leverages time domain diversity to fill-in the matching gap of existed 2D schemes. The primitive is to lengthen the transmission time of base-band data.

- TiM incorporates *Lengthen Coordinator* to reduce rate changing overhead.

- *Grain Size Estimation* module is designed for determining TiM's fine-grained level in wireless link to achieve better channel utilization.

- We implemented TiM on USRP2 and modified physical layer (PHY) preamble to make a hand-shake of TiM's interpolation factor between transceivers. Further, we validate and evaluate TiM's performance.

The rest of this paper is organized as follows. In section 2, we mainly discuss the motivation of TiM. Section 3 describes TiM's detailed system architecture. We implement TiM and assistant modules on USRP2 platform in Section 4. We evaluate TiM's performance in Section 5. Section 6 reviews the related work and Section 7 concludes the paper.

II. MOTIVATION: 3D MODULATION SCHEME WITH ADDITIONAL TIME DIMENSION

For many years, in information theory, there are basically three digital modulation types, namely PSK (Phase Shift Keying), ASK (Amplitude Shift Keying) and FSK (Frequency Shift Keying). PSK leverages phase shifting value to convey data, whereas ASK uses amplitude difference and FSK uses frequency shifting value. However, in IEEE 802.11 standards [4], basically, the modulation schemes (e.g. BPSK, 16-QAM) are all designed in the phase-amplitude two dimensions.

The reason why we do not leverage FSK or frequency domain schems is because the usage of OFDM (Orthogonal Frequency Division Multiplexing). OFDM is widely deployed in current WiFi communication system owing to its high efficiency of channel utilization. However, OFDM is not perfect. The shortage is that OFDM is strict with subcarries' orthogonality and time synchronizing. If we implement FSK into OFDM, it will decrease the synchronizing performance in OFDM. Further, Adding frequency domain modulation types also needs the offset estimation of doppler effect to be more precise. Because of these two main reasons, we cannot use FSK in current modulation schemes.

Even though each 2D modulation scheme (e.g. BPSK) has few coding rate (e.g. 1/2 3/4) options, it cannot achieve decent performance [14]. There are mainly two reasons. First, due to the limited number of pre-defined options, traditional 2D scheme cannot achieve varied fine-grained level modulation whereas TiM can. Second, TiM indeed can be implemented with any coding rate and make each of these options be more fine-grained which enhance the performance.

Because the non-perfect matching between existed modulation schemes and channel conditions, it causes throughput loss when changing modulation schemes. Nevertheless, conventional 2D modulation schemes have been studied for many years and have already been perfectly designed. So how

can we fill-in the matching gap without modifying the well-defined amplitude-phase domain schemes? Additionally, we cannot directly use the frequency domain schemes either.

Given all these concerns and analysis above, the only way we can modulate the signal besides amplitude-phase 2D domain is to design modulation schemes that leverage the time domain. This expansion from 2D to 3D is a way that we can fill-in the matching gap between current 2D modulation schemes and channel conditions.

III. TiM DESIGN

In this section, we describe the detailed design of TiM. Before presenting the system architecture, we first make a theoretical proof of why TiM can improve Packet Reception Rate (PRR) or throughput without changing the basic 2D scheme. Further, we discuss how well TiM can increase signal symbol's robustness by quantifying SNR gain that TiM can achieve in Additive White Gaussian Noise (AWGN) channel. Then we deliver the concrete design of TiM's modulation scheme and demodulation scheme separately.

A. Design Principles of TiM

In this part, we prove that TiM can improve PRR by strengthening signal's robust to noise and interference. We formally derive TiM's performance by modeling how the signal transmitted through the channel and then recovered at the receiver. We list some basic concepts and notations that we may use during the theoretical analysis and following sessions in table I.

Let $u[k]$ refers to the convolution value of the symbol emitted from sender $x_s[k]$ and the corresponding channel parameter $h[k]$. During the kth symbol transmission, the noise parameter is $n[k]$. The received symbol signal $y_r[k]$ is as equation 1,

$$y_r[k] = u[k] + n[k] = x_s[k] * h[k] + n[k] \tag{1}$$

Due to the limited pages, here we do not depict a figure of the receiver's CIC circuit, which is similar to the CIC circuit on the sender side in Fig.3. The difference between sender's and receiver's CIC circuit is just the a place exchange of integrator module and comb module (i.e. For the receiver's CIC circuit, on R module's left-hand side is integrator module, whereas on R module's right-hand side is comb module). Additionally, the R module in the receiver is used for decimation which is just the inverse operation of R module on the sender side. The received signal first pass through integrator module, then pass decimator (i.e. R module) and finally pass the comb module of CIC. Normally, in CIC circuit, we have integrator and comb with 4-stage as shown in Fig.3.

Definition 1 ($x_{iN}(k)$)**:** On the CIC circuit of the receiver side, let x_{iN} denotes the integrator module's output of the kth symbol passing through this circuit. The letter N refers to the number of stages in this integrator module, whereas letter i stands for integrator.

Definition 2 ($x_d(k)$)**:** Given the output of integrator x_{iN}, the data symbol need to pass through the decimator (i.e. R

TABLE I: Concepts and notations

N	Number of subcarries
R	Interpolation/Decimation factor of CIC circuit
$h[k]$	Channel parameters of the kth symbol
$n[k]$	Noise parameter of the kth symbol
$x_s[k]$	the kth signal symbol emitted from sender
$y_r[k]$	the kth signal symbol received at receiver side
ω_k	wight value for each data sample after pass CIC circuit
P	Signal transmission power
ξ_{snr}	SNR gain after we implementing modified CIC circuit

module). $x_d(k)$ denotes the decimator's output of kth symbol, where d represents the meaning of decimator.

Definition 3 ($y_{cN}(k)$)**:** Given $x_{iN}(k)$'s definition, here $y_{cN}(k)$ is quiet the same. Let y_{cN} denotes the signal output of comb module. The letter N share the same meaning as in $x_{iN}(k)$, whereas letter c stands for comb. The letter k represents the meaning that it is the kth symbol passing through this circuit.

There are mainly 3 components in the receiver's CIC circuit. We analysis this CIC circuit's output by calculating all its three parts separately. The output of the 4th integrator is as equation 2,

$$x_{i4}(n) = \sum_{k_1=0}^{n} \sum_{k_2=0}^{k_1} \sum_{k_3=0}^{k_2} \sum_{k_4=0}^{k_3} [u(k_4)] \tag{2}$$

we can get the output of decimator module as equation 3,

$$x_d(m) = \sum_{k_1=0}^{mR} \sum_{k_2=0}^{k_1} \sum_{k_3=0}^{k_2} \sum_{k_4=0}^{k_3} [u(k_4)] \text{ where } m \in [0, \lfloor \frac{n}{R} \rfloor] \tag{3}$$

After that, the signal should pass through 4-stage comb module. We can derive the final output of this whole 4-stage CIC circuit as equation 4,

$$y_{c4}(m) = \sum_{k_1=(m-4)R+4}^{(m-3)R+3} \sum_{k_2=k_1}^{k_1+R-1} \sum_{k_3=k_2}^{k_2+R-1} \sum_{k_4=k_3}^{k_3+R-1} u(k_4) \tag{4}$$

Given the noise parameter as $n(k)$, the output of noise y_n passing through CIC can be derived from the output of signal passing through CIC as equation 5.

$$y_n(m) = \sum_{k_1=(m-4)R+4}^{(m-3)R+3} \sum_{k_2=k_1}^{k_1+R-1} \sum_{k_3=k_2}^{k_2+R-1} \sum_{k_4=k_3}^{k_3+R-1} n(k_4) \tag{5}$$

Based on the output of CIC circuit (i.e. equation 4), we can get that the CIC output is the weighted (i.e. ω_k) sum of the last $4(R-1)$ input samples, which can be delivered as the equation 6,

$$y_s(m) = \sum_{k=(m-4)R+4}^{mR} \omega_k u(k) \tag{6}$$

Let's first calculate the SNR of wireless system without the CIC filter procedure. Suppose noise is modeled as AWGN, $n(k)$ meets with the Gaussian distribution. Thus we can derive that,

$$n(k) \sim N(0, \delta^2)$$

The SNR (i.e. ξ'_{snr}) can be represented as equation 7, where P is the signal transmission power,

$$\xi'_{snr} = \frac{E(u(k) - E(u(k)))^2}{E(n(k) - E(n(k)))^2} = \frac{P}{\delta^2} \tag{7}$$

When implementing CIC circuit with the interpolating factor of R at the sender side, the SNR gain (i.e. ξ_{snr}) at the receiver can be depicted as equation 8. *And we can easily derived that the SNR gain is larger than $R\frac{P}{\delta^2}$.*

$$\begin{aligned}\xi_{snr} &= \frac{E(y_s^2(m))}{E(y_n^2(m))} \\ &\geq \frac{R \sum_{k=(m-4)R+4}^{mR} \omega_k^2 E(u^2(k))}{\delta^2 \sum_{k=(m-4)R+4}^{mR} \omega_k^2} = R\frac{P}{\delta^2}\end{aligned} \tag{8}$$

Based on the proof above, we can derive that, *TiM's interpolating system can help the data signal to be more robust to noise and interference.* The more interpolated values between adjacent data points, the better recovery of data points at the receiver side.

B. 3D Modulation/Demodulation Schemes

To realize TiM, there are several challenges. First, at the sender side, the lengthen process of base band data must overcome the following difficulties: (1) it must be configurable to set different lengthen value during the rate adaptation process. (2) how to determine the lengthen value with varied channel condition is an open problem. We deliver our methods to deal with these two problems from both theoretical analysis and Matlab simulation results.

On the receiver side, the design of demodulation scheme can be more challenging. The first issue is the delay of changing interpolation/decimation rate between sender and receiver. We present *Lengthen Coordinator* to solve this problem. Another difficulty is how to recover data points with less accurate coordination and synchronization between the sender and the receiver. To handle this problem, we leverage the interpolated points to help improve the accuracy of data recovery.

1) Modulation Scheme Design: The key idea of the Timeline modulation scheme is to lengthen the transmission time of base-band data. To do so, we need to interpolate redundancy values between any two adjacent original data points we need to transmit. As illustrated in Fig.2, we modified the CIC circuits on sender side's low pass interpolator (LPI) [8] in order to change interpolation factor dynamically. With this modified LPI, based on the estimation of current SNR, we can decide how many interpolation values we need to insert between adjacent real data points.
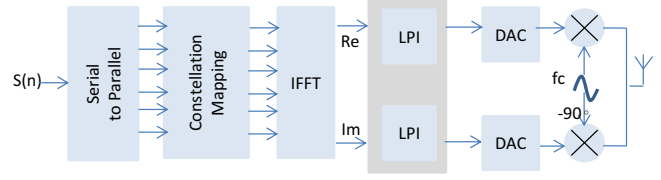

Fig. 2: Sender side OFDM system structure with Low Pass Interpolation (LPI) block. TiM modified its CIC (Cascaded Integrator-Comb) filter to adjust interpolation parameter.
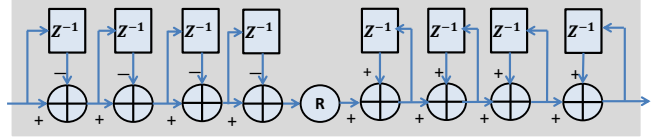

Fig. 3: Inner structure of TiM's Modified CIC circuit in sender's LPI block.

First we make an analysis about this modification in order to illustrate this process more clearly. Let the data symbols we want to transmit be the $\{X[k]\}$ set. For each data symbol $x[n]$, the output of IFFT (Inverse Fast Fourier Transform) is as equation 9,

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k]e^{j2\pi\frac{kn}{N}} \tag{9}$$

In Fig.2, after the IFFT process at the sender side, the plural data symbol $x[n]$ we need to transmit is separated into two flows, namely real part (Re) and imaginary part (Im). After that, each of these two symbol flows (i.e. Re x[n] and Im x[n]) are independently passing through the LPI module with an interpolation factor. Here we use R to denote the interpolation factor.

As shown in Fig.3, the CIC filter here is specially designed. It contains a 4-stage cascaded integrators, an interpolation module and a 4-stage comb. We modified the CIC filter in order to make the interpolating factor can be reprogrammed. The main duty of this module is used for interpolation by using Z-domain transformation. The left-hand side which is before module "R" in Fig.3, it is called comb module. Let $y_1[z]$ and $x_1[z]$ denote the output and input of this part circuit, respectively. The Z-domain transfer function $H_C(z)$ is illustrated as equation 10,

$$H_C(z) = \frac{y_1(z)}{x_1(z)} = (1 - z^{-1})^4 \tag{10}$$

Integrator module is on the right-hand side of module "R". $y_2[z]$ and $x_2[z]$ also denote the output and input of this part circuit, respectively. The corresponding Z-domain transfer function $H_I(z)$ is as equation 11,

$$H_I(z) = \frac{y_2(z)}{x_2(z)} = \left(\frac{1}{1 - z^{-1}}\right)^4 \tag{11}$$

The function of the modified "R" module is to do interpolation process, which can be represented as equation 12. Here $y_3[z]$ and $x_3[z]$ also denote the output and input of this part circuit.

$$y_3[n] = \begin{cases} x_3[\frac{n}{R}] & \text{if } n = kR \\ 0 & \text{other} \end{cases} \quad (12)$$

One point needs to be explained. The value we artificially interpolated between the data sampling points has the initial value of 0. Because of the special characteristic of the CIC filter, the interpolated 0-value points can be changed into redundancy values which are helpful for recovering data at the receiver side. For more details, we suggest to read [15] and [13] which have detailed theoretical explanation of CIC filter technique.

Based on these basic equations, We can derive the whole circuit's Z-domain transfer function as equation 13,

$$\begin{aligned} H(z) &= H_c(z^R) \times H_I(z) \\ &= (1 - z^{-R})^4 \times \left(\frac{1}{1 - z^{-1}}\right)^4 \\ &= \left(\frac{1 - z^{-R}}{1 - z^{-1}}\right)^4 \end{aligned} \quad (13)$$

According to this Z-transfer function in equation 13, we transfer it into time-domain and get the input $x[n]$ and output $y[n]$ of this whole modified LPI as equation 14,

$$\begin{aligned} y[n] &= 4y[n-1] - 6y[n-2] + 4y[n-3] - y[n-4] \\ &\quad + x[n] - 4x[n-R] + 6x[n-2R] - 4x[n-3R] \\ &\quad + x[n-4R] \end{aligned} \quad (14)$$

Based on equation 14, *we can change the length of each data symbol's transmission time by interpolating values between adjacent data points.* Furthermore, the artificially added redundancy points will help the signal to be more robust to interference and noise during transmission and can be better recovered at the receiver, which has been proved in section 3.1. To make it clear, the whole process of interpolating 0-value points and finally get the redundancy values is represented as Fig.4.

Another challenge we need to conquer is how to set the specific lengthen value to deal with varied channel condition. Based on Shannon's theorem, we first make a theoretical analysis about the relationship between the lengthen value and signal's resistance to noise and interference.

From the information theory, the channel capacity (i.e. C) can be represent as equation 15,

$$C = B \log_2(1 + \frac{S}{N}) \quad (15)$$

Additionally, we can derive that, for a given SNR with one fixed modulation scheme, we have the relationship between highest throughput (i.e. T) and channel capacity as depicted in equation 16,

(a) Original data points.

(b) Original data points with artificial interpolated points.

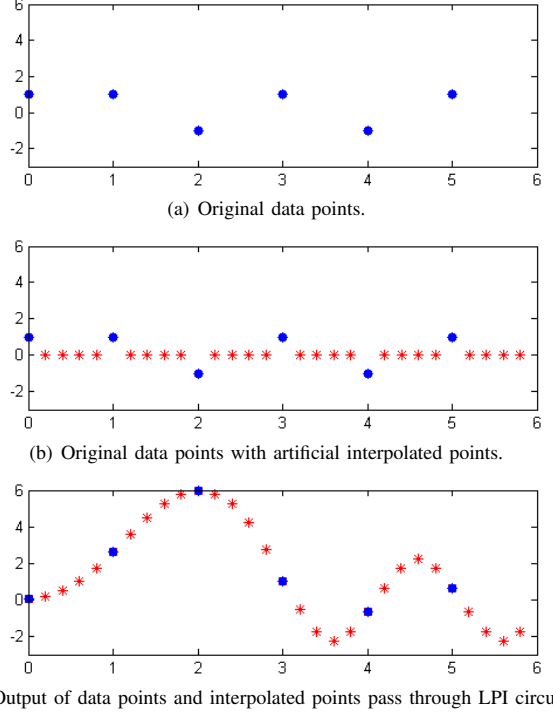(c) Output of data points and interpolated points pass through LPI circuit.

Fig. 4: The process of interpolating points between adjacent data points pass through LPI circuit, where blue points stand for original data points and red stars denote interpolation points.

$$T \propto C \quad (16)$$

Based on this, since the interpolation factor R is inversely proportional to the throughput T, we can derive the relation between interpolation factor R and $\frac{S}{N}$ as equation 17, where T_{new} denotes the throughput of changed interpolation scheme with new interpolation value R_{new}. And T_{ori} stands for the original throughput with the initial interpolation value R_{ori}. S_{new} is the signal strength of changed interpolation factor whereas S_{ori} is signal strength with the original interpolation factor.

$$\frac{R_{ori}}{R_{new}} = \frac{T_{new}}{T_{ori}} = \frac{C_{new}}{C_{ori}} = \frac{B \log_2(1 + \frac{S_{new}}{N})}{B \log_2(1 + \frac{S_{ori}}{N})} \quad (17)$$

Now we can draw the conclusion that, *given a fixed bandwidth, R is inversely proportional to $log_2(1 + \frac{S}{N})$.* In other words, *given one fixed bandwidth, the more interpolation points being inserted, the lower SNR needed for the receiver to correctly recover the real data symbols.* Further, our Matlab simulation result in Fig.5 tests and verifies our theoretical analysis. Based on this relationship, we can map the specific lengthen value (i.e. interpolation factor) to the varied channel conditions according to the mapping rules depicted in Fig.5.

2) Demodulation Scheme Design: In TiM's demodulation process, it is not just a reverse procedure of the modulation
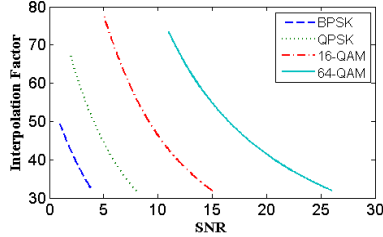
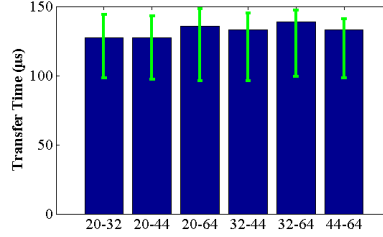Fig. 5: Mapping rules for interpolation/decimation factor with varied SNR.

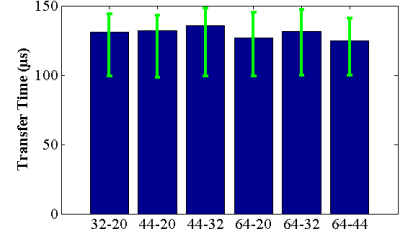Fig. 6: Time delay of increasing interpolation.

Fig. 7: Time delay of decreasing interpolation.

process. The design of TiM's demodulation scheme seems to be more challenging. And there are mainly two following difficulties.

First, when changing the interpolation rate, the coordination between transceivers is a hard problem. How to manage this procedure in order to minimize the overhead of rate changing delay? The module "*Lengthen Coordinator*" we proposed in section 4 can solve this problem with negligible overhead.

Second, when SNR varies frequently, our interpolation rate changing may also be frequent. Given this, the receiver's sampling rate may not be perfectly synchronized and coordinated with the sender's. There may exists difference between the original data points and receiver's sampled values. Own to our interpolation points, this distortion can be recovered by TiM's interpolated value. This recovery procedure can be regarded as another aspect of enhancing signal's robust attribute that TiM can achieve.

IV. IMPLEMENTATION

In this section, we present a detailed implementation of TiM. We use GNU radio [1] to implement TiM's sender and receiver on USRP2 software radio platform [2]. As depicted in Fig.8, TiM's transceivers mainly consist of 3 parts, namely *Grain Size Estimation*, TiM modulation/demodulation and *Lengthen Coordinator*. The main working procedure is as follows. First, with channel variation information, by using *Grain Size Estimation*, we set proper fine-grain level to balance between the overhead of changing rate and fine-grained modulation gain. Then, we implement TiM modulation/demodulation for upper layer data encoding/decoding and leverage *Lengthen Coordinator* to reduce the rate changing head.

In following parts, We first deliver the implementation procedure of the TiM's modulation/demodulation design on the sender and the receiver. After that we discuss about *Lengthen Coordinator* and *Grain Size Estimation* design.

A. TiM's Modulation/Demodulation Scheme Implementation

As described in section 3.2, we need to modify module "R" in the CIC circuit of transceivers to enable that the interpolation/decimation factor can be reprogrammed. After this modification, we artificially insert the interpolation points (with initial value of 0) before the interpolation process. Then we can change the interpolation factor based on the number
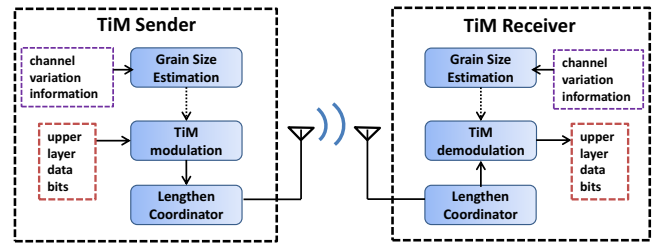

Fig. 8: High-level architecture of TiM's sender and receiver.

of interpolation points we insert between two adjacent real data points. We let the data points and inserted interpolation points together pass through the CIC circuit and finally get the combined modulated signal as the CIC's output. Then we transmits the modulated signal to the receiver.

On the receiver side, we demodulate the combined signal as an inverse process of the modulation procedure on the sender side. After we decoded one specific data point and its correlated interpolation values, we leverage the interpolation values to make an estimation of this data point to enhance the accuracy of data recovery. Since there are huge amounts of value estimation algorithms (such as belief propagation in [3], confidence level in [11] and so on), here we use the K-means clustering algorithm [9] to do this estimation.

B. PHY Preamble Modification

In order to coordinate the transceiver's interpolation/decimation rate, we need to modify PHY preamble. In IEEE 802.11 standard [4], modulation scheme information is contained in the RATE field of OFDM PLCP (Physical Layer Convergence Protocol) preamble.

The RATE field of OFDM PLCP preamble consists of 4 bits, which can represent at most 16 modulation schemes. The current modulation schemes in use are BPSK QPSK QAM16 with coding rate of 1/2 and 3/4, and QAM64 with coding rate of 2/3 and 3/4. Therefore, there are $16 - 8 = 8$ empty positions can be used for representing additional modulation schemes provided by TiM. As for normal situations, we believe that adding 8 modulation schemes is fine-grained enough to get the near optimal goodput in a wireless link. So here we leverage these 8 available positions in RATE field to represents interpolation factor of 8 additional TiM's modulation schemes.

Furthermore, given more fine-grained scenarios, we can modify PHY preamble by adding additional bits on RATE field to represent a larger number of more fine-grained modulation schemes in TiM.

C. Lengthen Coordinator

We now present the design of *Lengthen Coordinator* module, which is used for reducing the overhead of rate changing. Basically, there are two main changing overheads. One is the delay in the rate (interpolation factor) changing process. Another is the delay of processing received data at the receiver side. We first discuss these two delays respectively. Then we present *Lengthen Coordinator*'s scheduling algorithm to solve these two issues.

1) Delay in Changing Interpolation Rate: When changing the transmission time of each data symbol, the radio need to be stabilized before transmitting or receiving signals. According to our experimental results in Fig.6 and Fig.7, the average changing delay is nearly 130us. More precisely, the delay of changing rate varies from 118us to 140us. In addition, Fig.6 and Fig.7 also depict that changing rate between two specific interpolation factors has different delay. For example, the delay of increasing interpolation factor from 44 to 64 is longer than the delay of decreasing from 64 to 44. Given this delay issue, We need to manage this delay time, so that it has less probability of disturbing the current wireless transmission during the changing rate process.

2) Delay of Received Signal Processing: On the other side, there is another interesting problem. When we let the sender change the interpolation rate after sending the 1000th packet, however, every time the receiver can only decode the first 994 packets. So why the last six packets cannot be decoded correctly?

To answer this question, we change the interpolation rate changing time of the sender. First, we set the rate changing time after the sender transmits the 500th packet, then after the 1000th, and then after the 2000th packet, so on and forth. The quantity of packets that cannot be decoded is the same. We measure the time point of sender sending packets and receiver receiving packets. The data in table II shows the time of emitting and receiving packets where sender changes rate on the 501th packet.

In table II, there is a time gap between the sender sending one packet and the receiver receiving that specific packet. This is because processing delay on both sender and receiver (e.g. packet 494, the delay between sender and receiver is $28.43 - 28.35 = 0.08ms$). And this is the key reason why receiver cannot decode the last 6 packets with unchanged interpolation rate.

As in table II, when the receiver finish processing the 494th packet, it is the same time that the sender sends the 501th packet. It means that the receiver's buffer has already received the first 500 packets but not finished processing them. And at the same time, it receives the 501th packet. Since interpolation rate is not the same as the first 500 packets, the receiver cannot decode the 501th packet. Because of rate changing, receiver stops processing and empties the buffer that stores unprocessed packets (i.e. 495-500). After that, the receiver received the 502th packet and decoded it as normal.

TABLE II: Packet transmission delay between transmitter and receiver

Packet number	Sender transmit time (ms)	Receiver received time (ms)
494	28.35	28.43
495	29.09	N/A
496	29.64	N/A
497	30.18	N/A
498	30.76	N/A
499	31.37	N/A
500	32.03	N/A
501	32.61	N/A
502	33.18	33.26

3) Lengthen Coordinator Scheme Design: With the two main delays we discussed above, how to reduce the overhead is a big challenge. *Lengthen Coordinator* is a tricky scheme. Instead of directly avoiding these overheads, it leverage these two delays to cancel with each other.

To illustrate this, we also use table II as reference. After the sender emitted the 500th packet with original interpolation rate, it finished its rate changing process in around 130us. At this time, the receiver finished processing the 494th packet. Because the delay between sender and receiver is caused by both sides, it needs nearly half of the total delay time (i.e. $0.08/2 = 0.04ms$) for the receiver to process one packet. Thus the receiver needs $0.04ms \times 6 = 240us$ to finish processing the last 6 packets in its buffer. Given this, the sender prepares the 501th packet and doesn't send it, this period costs 40us. At this time the receiver has processed packet 495. The sender waits for another $0.04ms \times 5 = 200us$ which is the time for the receiver to finish processing from the 496th packet to the 500th packet. After this, the sender begins to send the 501th packet with changed rate. The receiver received this packet and began to coordinate with the sender's current interpolation rate. So there is only one packet (i.e. the 501th packet) loss rather than 7 (i.e. the 495th-500th packets with original rate and the 501th packet with changed rate) loss. Additionally, at the receiver side, there is no noticeable delay because the receiver received the following packet (i.e. the 501th packet) right after it processed the packets (i.e. the 495th-500th packet) in its buffer. In receiver's view, it is as if the transmission has no delay since it always has packets received need to be processed.

By leveraging processing delay and rate changing delay to cancel each other, we successfully reduce the overhead of implementing TiM into real-world systems.

D. Grain Size Estimation

Only based on the basic model of TiM, it cannot be implemented into real-world wireless networks. Before implementing TiM, we should first define how dense the fine-grained level we set. *Grain Size Estimation* can achieve this by measuring channel variation range and frequency.

Before implementing TiM, we first make a training process of channel condition estimation. Based on the channel variation information we collect, we analysis its variation range and frequency. The basic principle is that the larger variation range and the higher variation frequency, we set less fine-grained levels (and vice versa) to meet with the trade-off between rate changing overhead and fine-grained throughput gains.
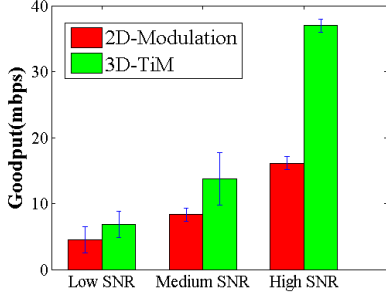
Fig. 9: Goodput comparison between traditional 2D modulation schemes and 3D TiM with varied SNR.
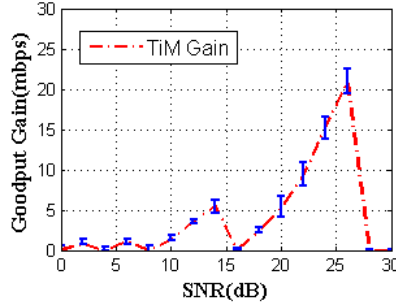

Fig. 10: TiM's goodput gain compared with rate adaptation using traditional 2D modulation schemes.
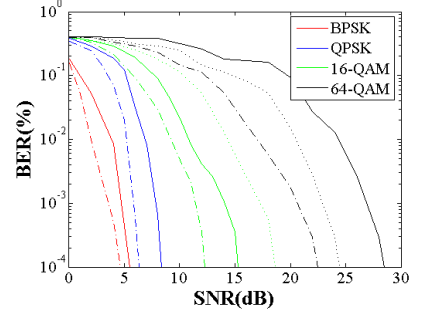

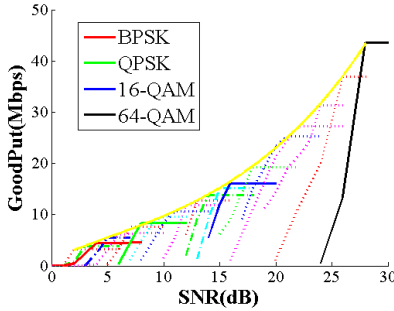Fig. 11: TiM's BER performance with varied SNR.

Fig. 12: TiM's nearly linear rate changing performance with varied SNR.
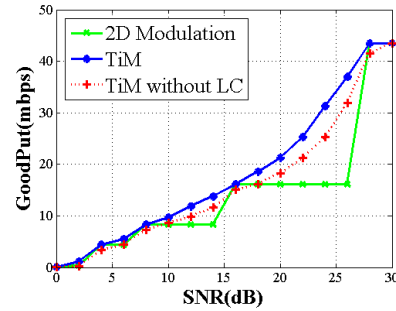

Fig. 13: Performance of TiM with/without Lengthen Coordinator(LC) in varied SNR.
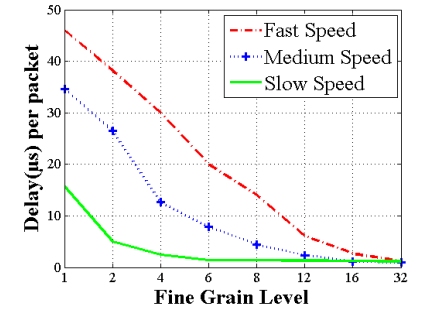

Fig. 14: TiM's fine-grained level changing delay.

Additionally, after implementing TiM, we can further adapt fine-grained level dynamically if the channel condition changes dramatically.

V. EVALUATION

We evaluate TiM's performance in this section. Here we mainly focus on channel utilization, *Lengthen Coordinator* effect, fine-grain level effect and the transceivers' moving effect. The experimental results demonstrate that TiM can achieve up to 200% channel utilization efficiency. Furthermore, without *Lengthen Coordinator* the overall goodput will decrease. Additionally, *Gain Size Estimation* have a significant impact on the performance of TiM.

A. Channel Utilization

In this part, we evaluate TiM's performance mainly from three aspects, namely the channel utilization, bit error rate (BER) and how the nearly linear rate changing that TiM can achieve.

First, we compare the goodput (not throughput) of TiM with conventional 2D modulation in three different scenarios, namely Low SNR (ranging from 0-10 dB), Medium SNR (10-20 dB), High SNR (20-30 dB). Note that each 2D scheme (e.g. BPSK) may have different coding rates (e.g. 1/2 3/4), here we pick up the coding rate that can achieve the highest goodput as the coding rate for this specific 2D scheme. And we use the highest goodput to represent each scheme's performance.

In addition, the performance of all the 2D schemes in Fig.9, Fig.10, Fig.12 and Fig.13 follows the same rules.

As shown in Fig.9, in low and medium SNR, TiM outperforms 2D schemes to nearly 60% on average. In high SNR, TiM achieves up to 200% goodput compared with 2D schemes.

To make it clear, Fig.10 depicts TiM's gain over traditional 2D schemes with the SNR ranging from 0-30 dB. The gain achieves its peak value in the SNR interval of two adjacent 2D schemes. Further, the peak value increases from sparse schemes (e.g. BPSK) to dense schemes (e.g. 64-QAM). It means that the denser 2D modulation scheme we use, the larger gap will exist in rate adaptation. This result also exposes the defect of matching gap between 2D schemes and continuous varied channel condition. TiM can cover up this gap to improve channel utilization.

Second, we get BER result with varied SNR in the experiment. In Fig.11, compared with the BER-SNR result of traditional 2D schemes, TiM makes it denser and become more fine-grained. It means that TiM can not only fill-in the goodput gap of traditional schemes, but also cover up their BER gap. In addition, Fig.11 also verifies that TiM's data lengthen process (i.e. interpolation process) indeed enhances the signal's robust to noise and interference.

Last, we test with TiM's modulation process to achieve nearly linear rate changing in rate adaptation. In USPR2 [2] experimental environment, we make the highest fine-grained level we can. We can see from Fig.12 that we add
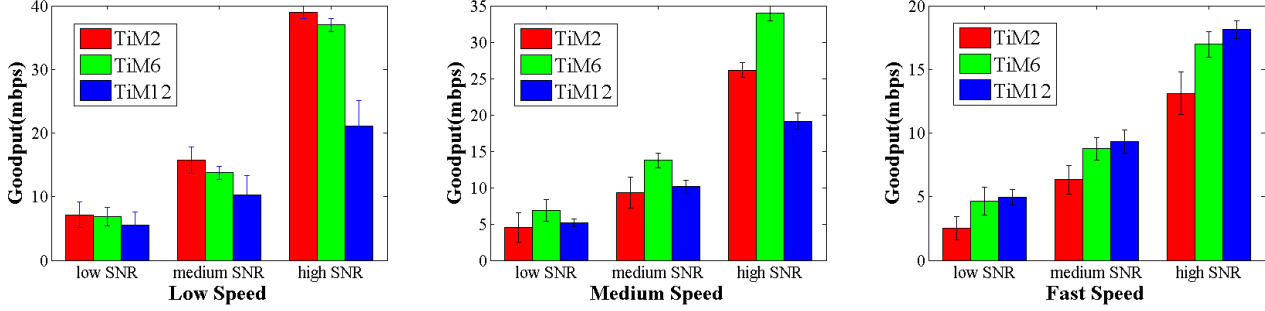
584

Fig. 15: Moving speed effects on TiM's performance with different fine-grained level and varied SNR.

5 intermediate TiM's schemes on average between each two adjacent traditional 2D modulation schemes. The results in Fig.12 also shows that, with TiM's fine-grained modulation schemes, the rate adaptation process is no longer staircase-like, which has gap when changing from one bitrate modulation scheme to adjacent ones. On the contrary, rate adaptation can achieve nearly linear bitrate adjustment with TiM's fine-grained modulation schemes. There is no noticeable goodput loss during this fine-grained modulation scheme changing process. In other words, results in Fig.12 verifies that TiM can achieve the mapping process between TiM's 3D modulation schemes and the estimated continuous SNR value without noticeable performance loss.

B. Lengthen Coordinator Effect

As depicted in Fig.13, there exists difference between the performance of TiM with/without *Lengthen Coordinator* (LC). In some circumstances, the goodput performance of TiM without LC really drops a lot which is lower than traditional 2D modulation schemes. Without dealing with the two main kinds delays properly will cause significant loss of TiM's overall goodput.

However, as a whole, TiM without LC can still outperform over the traditional 2D schemes because that the fine-grained gain of TiM is larger than delay overhead in most of the circumstances. We can deduce that sometimes if we cannot implement LC in real-world scenarios, TiM can still achieve better performance than traditional 2D schemes in most of the circumstances.

C. Grain Size Effect

We evaluate the module *Grain Size Estimation*'s effect on TiM's performance. We mainly focus on two aspects. First, we discuss about the time delay that different fine-grained levels will cause. The prime results show that the more fine-grain level we set, the higher delay will cost. Second, we explore the nodes' movement effect on different fine-grain level schemes. We find out that the coarser grain level we use, the more robust to the moving scenarios and vice versa.

We measure the time delay of different fine-grained levels with the same quantity of packets for transmission. And we calculate the average time delay per packet. In Fig.14, the number on horizontal axis represents the fine-grain level of TiM. For example, if the value on horizontal axis of Fig.14

is 1, it means that the minimum rate changing unit is 1 (i.e. suppose the original interpolation factor is 6, with fine-grain level of 1, the factor can be changed to 5,7 and so on). From Fig.14, as a whole, we can find out that the higher fine-grained level we set, the more delay overhead will cause because of the frequently rate changing.

We also test the delay with transceiver's movement of different speeds. As for normal walking or little running is within 1-9 mph [3], we evaluate different fine-grain level with moving speed ranging from 1-9 mph. More precisely, we divide the moving speed to three level, namley low speed (1-3 mph), medium speed (4-6 mph) and high speed (7-9 mph). The speed specification of Fig.15 follows the same rule. As shown in Fig.14, with the assistance of LC, the delay overhead decreases to a relatively acceptable value when we use fine-grained level coarser than 6 in all three kinds of moving speed. Given this, we can deduce that typically the changing unit of the interpolation factor can be set to 6 with acceptable overhead when facing with varied moving speeds.

Fig.15 illustrates the varied moving speeds impact on TiM from the goodput perspective. It depicts detailed experimental results of TiM's goodput performance with different fine-grained level TiM in varied SNR and different moving speed. As shown in Fig.15, even though the high fine-grained (i.e. TiM2 with minimum rate changing unit of 2) scheme has higher performance than others when the transceivers are relatively stable (i.e. low moving speed and high SNR), the overall performance of the medium fine-grained scheme (i.e. TiM6 with minimum rate changing unit of 6) is the best. This result also verified that we need to make a trade-off between the gain of the TiM's fine-grained level and the overhead caused by frequently rate changing.

VI. RELATED WORK

TiM is closely related to two kinds of literature. The first kind is rate adaptation based on 2D modulation schemes. The second category is to improve channel utilization by adapting spectrum width or using time interval to convey data.

A. Rate Adaptation Based on 2D Modulation Schemes

Rate adaptation in wireless networks has been a hot research topic for long time. There are a lot of papers addressing rate adaptation problems. Most of recent works can be classified into loss-based schemes and SNR-triggered ones.

Many papers estimate channel condition based on historical loss information [17] [12]. *softrate* [21] uses BER of received packets to estimate bitrate that channel can support. *AccuRate* in [19] leverages channel distortion to estimate the best rate. The authors propose *RRAA* in [22] that leverages loss information in short frame window to estimate channel condition. *COLLIE* in [17] allows the receiver to send back error packets to the sender in order to diagonalize the cause of the error.

SNR-based protocol is another estimation method for rate adaptation. Many current works focus on evaluating channel SNR to estimate bitrate [24] [10] [18] [16]. In [16], it demonstrates to use different rate adaptive modulation schemes with varied frequency band. In [5], Camp et al. propose a framework to evaluate rate adaptation schemes. And its conclusion is that trained SNR-based protocols outperform loss-based ones.

However, all of them are based on existed 2D modulation schemes. Thus the matching gap between modulation types and varied channel conditions still causes throughput loss. Only changing the modulation schemes into TiM's 3D domain will fix with this throughput loss.

B. Channel Utilization

Channel Width Adaptation: Some recent papers focus on how to utilize channel more efficiently [6] [20]. *FICA* in [20] proposes to divide a wide band into separated narrow bands which can improve channel utilization. On the other hand, In [6], the authors claim to use varied channel width for different bitrate transmission. However, since they all narrow down width of channel bands, there will be more guard bands which decrease the channel utilization. On the contrary, TiM indeed will not cause spectrum width changing. Furthermore, the signal's robust does not improve as what TiM can achieve.

Time Interval Modulation: There is another kind of approaches that instead of avoiding interference, they leverage interference to convey control information [7] [23]. In their proposals, they leverage the time interval between any two adjacent transmitted symbols to convey information. However, this kind of time domain usage does not improve signal's robust to interference and noise whereas TiM does.

VII. CONCLUSION

In this paper, we propose a novel 3D modulation scheme TiM with additional time domain. It adds time-domain into existed modulation scheme's 2D (amplitude and phase) domain in order to fill-in the matching gap between limited modulation types and continuous varied channel conditions. Our measurement shows that TiM can achieve nearly linear rate changing during rate adaptation instead of current scheme's staircase-like performance. This makes TiM improve channel utilization up to 200% and increase signal's robust to noise and interference. Furthermore, assistance of *Lengthen Coordinator* and *Grain Size Estimation* enhances TiM's performance. Since TiM follows traditional modulation schemes, we believe TiM can be beneficial to widely deployed commercial WLANs.

ACKNOWLEDGEMENTS

REFERENCES

[1] GNU software defined radio, http://www.gnu.org/software/gnuradio.

[2] *Universal Software Radio Peripheral*. Ettus Research LLC, http://www.ettus.com.

[3] S. T. Aditya and S. Katti. Flexcast: Graceful wireless video streaming. In *Proceedings of ACM MobiCom*, 2011.

[4] E. Blossom. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE Std 802.11, 2012.

[5] J. Camp and E. Knightly. Modulation rate adaptation in urban and vehicular environments: Cross-layer implementation and experimental evaluation. In *Proceedings of ACM MobiCom*, 2008.

[6] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghhavendra, and P. Bahl. A case for adapting channel width in wireless networks. In *Proceedings of ACM SIGCOMM*, 2008.

[7] A. Cidon, K. Nagaraj, S. Katti, and P. Viswanath. Flashback: Decoupled lightweight wireless control. In *Proceedings of ACM SIGCOMM*, 2012.

[8] M. E. Frerking. *Digital Signal Processing in Communication Systems*. Kluwer Academic Publishers, Norwell, MA, USA, 1993.

[9] J. Han and M. Kamber. *Data Mining: Concepts and Techniques (2nd Edition)*. Morgan Kaufmann Publishers, San Francisco, CA, USA, 2006.

[10] G. Holland, N. Vaidya, and P. Bahl. A rate-adaptive mac protocol for multi-hop wireless networks. In *Proceedings of ACM MobiCom*, 2001.

[11] K. Jamieson and H. Balakrishnan. PPR: Partial packet recovery for wireless networks. In *Proceedings of ACM SIGCOMM*, 2007.

[12] J. Kim, S. Kim, S. Choi, and D. Qiao. Cara:collision-aware rate adaptation for ieee 802.11 wlans. In *Proceedings of IEEE INFOCOM*, 2006.

[13] J. Y. Kwentus, Z. Jiang, and A. N. Willson. Application of filter sharpening to cascaded integrator-comb decimation filters. *IEEE Transactions on Signal Processing*, 45(2), 1997.

[14] K. C.-J. Lin, N. Kushman, and D. Katabi. Ziptx: Harnessing partial packets in 802.11 networks. In *Proceedings of ACM MobiCom*, 2008.

[15] R. G. Lyons. *Understanding Digital Signal Processing (2nd Edition)*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2004.

[16] H. Rahul, F. Edalat, D. Katabi, and C. Sodini. Frequency-aware rate adaptation and mac protocols. In *Proceedings of ACM MobiCom*, 2009.

[17] S. Rayanchu, A. Mishra, D. Agrawal, S. Saha, and S. Banerjee. Diagnosing wireless packet losses in 802.11:separating collision from weak signal. In *Proceedings of IEEE INFOCOM*, 2008.

[18] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly. Opportunistic media access for multirate ad hoc networks. In *Proceedings of ACM MobiCom*, 2002.

[19] S. Sen, N. Santhapuri, R. R. Choudhury, and S. Nelakuditi. Accurate: Constellation based rate estimation in wireless networks. In *Proceedings of NSDI*, 2010.

[20] K. Tan, J. Fang, Y. Zhang, S. Chen, L. Shi, J. Zhang, and Y. Zhang. Fine-grained channel access in wireless lan. In *Proceedings of ACM SIGCOMM*, 2010.

[21] M. Vutukuru, H. Balakrishnan, and K. Jamieson. Cross-layer wireless bit rate adaptation. In *Proceedings of ACM SIGCOMM*, 2009.

[22] S. H. Wong, H. Yang, S. Lu, and V. Bharghavan. Robust rate adaptation for 802.11 wireless networks. In *Proceedings of ACM MobiCom*, 2006.

[23] K. Wu, H. Tan, Y. Liu, J. Zhang, Q. Zhang, and L. M.Ni. Side channel: Bits over interference. In *Proceedings of ACM MobiCom*, 2010.

[24] J. Zhang, K. Tan, J. Zhao, H. Wu, and Y. Zhang. A practical snr-guided rate adaptation. In *Proceedings of IEEE INFOCOM*, 2008.