Practical and Efficient Open-Loop Rate/Link Adaptation Algorithm for High-Speed IEEE 802.11n WLANs[1]

Qiuyan Xia, Jian Pu, Mounir Hamdi
Department of Computer Science and Engineering
Hong Kong University of Science and Technology
Kowloon, Hong Kong
{xiaqy, pujian, hamdi}@cse.ust.hk

Khaled Ben Letaief
Department of Electronic and Computer Engineering
Hong Kong University of Science and Technology
Kowloon, Hong Kong
eekhaled@ece.ust.hk

***Abstract*—In this paper, we propose a new open-loop rate/link adaptation algorithm (ARFHT) for the emerging high-speed IEEE 802.11n WLANs. ARFHT extends the legacy rate adaptation algorithms for SISO WLANs to make it applicable in the context of MIMO-based 802.11n WLANs. It adapts the MIMO mode in terms of spatial multiplexing and spatial diversity, the two fundamental characteristics of the 802.11n MIMO PHY. It also modifies the link estimation and probing behavior of legacy SISO algorithms. The combined adaptation to the appropriate MIMO mode as well as the appropriate modulation coding scheme selection achieves high channel utilization. In this paper, we provide the intuition and the design details of the ARFHT algorithm. A comprehensive simulation study using ns-2 will demonstrate that ARFHT achieves excellent throughput performance in most scenarios and is highly responsive to varying link conditions, with minimum overhead.**

I. INTRODUCTION

With the growing demands for faster and higher capacity WLANs, the IEEE 802.11 Task Group-n (TGn) seeks to achieve higher physical layer data rates and improved MAC efficiency in the next generation WLAN standard 802.11n. It will be backward-compatible with 802.11a/b/g, and will improve the peak throughput to at least 100 Mbps measured at the MAC data SAP (Service Access Point) [1]. The pre-1.0 draft of 802.11n standard, which is largely a proposal of the Enhanced Wireless Consortium [2] [3] with some changes, has been approved in January 2006. It defines a HT-PHY based on the MIMO-OFDM technologies. OFDM is well suited for transmission in frequency-selective fading channels. The use of multiple transmit and receive antennas at the stations, or MIMO (Multiple Input Multiple Output), provides spatial diversity to improve the range (or alternatively, reliability) and spatial multiplexing for higher data rates.

Similar to the SISO (Single Input Single Output) WLANs, 802.11n specifies a set of Modulation Coding Schemes (MCS), each regulating the modulation coding used in its spatial streams. The actual transmission scheme can be adaptively selected to suit the MIMO channel conditions. When transmitting a certain number of spatial streams, higher level MCS requires higher SNR (Signal to Noise Ratio) to maintain a small BER (Bit Error Rate). Besides, the MIMO PHY supports two modes of operation: spatial diversity for better signal quality and spatial multiplexing for higher throughput. These two MIMO modes can be adjusted by sending data over variable number of spatial streams (N_{SS}). When N_{SS} is set to smaller values than the number of transmit antennas N_{TX}, the additional antennas can be exploited for diversity gain. The observation is that we need to use multiplexing at higher SNR regions as long as packet error rates are small, and have mechanisms to detect persistent channel deteriorations and switch to diversity to increase reliability at lower SNR regions.

Considering the time-varying link qualities due to multi-path fading, movements of surrounding objects and so on, selection of a suitable combination of MCS and MIMO mode is critical to the overall system performance. The 802.11n proposal supports both open-loop and closed-loop rate/link adaptations. The closed-loop operation assumes that some channel knowledge is available at the transmitter, either through explicit feedback from the receiver using specific control frames, or through channel sounding and calculation between the transmitter and the receiver. However, the computation complexity and the communication overhead incur implementation difficulties.

In this project, we focus on developing an open-loop rate/link adaptation algorithm at the MAC layer, based on the most recent PHY proposal for 802.11n. The algorithm should be highly adaptable to a variety of wireless environments that may change in a very short time. It requires no complicated channel state calibration at the transmitter, and no communication overhead is involved since only the transmitter's statistics are utilized. Moreover, the algorithm is transparent to the data communications protocols. Therefore, it can be integrated with existing systems with little or no change to the latter. Finally, this open-loop extension is particularly important in heterogeneous 802.11 WLANs, for maintaining seamless interoperability and coexistence with legacy devices, which typically, only utilize open-loop link adaptation.

In the proposed ARFHT algorithm, the transmission scheme is mainly determined according to previous transmission history. However, it is now a two-dimensional adaptation: the search for the appropriate MCS of all spatial streams, and the MIMO mode leveraging spatial diversity and/or spatial multiplexing. Specifically, we first derive a relationship that allows the transmitter to estimate the channel quality dynamics, observing the link layer acknowledgement (ACK) and the Received Signal Strength Indicator (RSSI). Intuitively, this relationship should account for the transmitter's "credits" accumulated in previous transmissions for its future probing. To do so, we maintain several statistic counters which are dynamically updated. Then the search behavior is determined based on the predicted channel quality dynamics: the vertical search regarding the MCS adjustment and the horizontal search for N_{SS} adjustment (Table 1). We propose a novel link probing method accounting for both dimensions. We then design the open-loop link adaptation rule: basically, this is a threshold-based scheme with the goal of throughput maximizing (or alternatively, minimizing the expected

---

[1] This work was supported in part by a grant from the Research Grants Council (RGC) under contract HKUST610307.

transmission time). MCS is adapted when one or more counters exceed the corresponding thresholds. The main concern is to make all parameters adaptive to instantaneous channel conditions.

Table 1. High-Throughput Basic MCS for Mandatory 20 MHz Modes

RATE \ N_{SS}	1	2	3	4
6.5x BPSK 1/2	0 (6.5)	8 (13.0)	16 (19.5)	24 (26.0)
13.0x QPSK 1/2	1 (13.0)	9 (26.0)	17 (39.0)	25 (52.0)
19.5x QPSK 3/4	2 (19.5)	10 (39.0)	18 (58.5)	26 (78.0)
26.0x 16QAM 1/2	3 (26.0)	11 (52.0)	19 (78.0)	27 (104.0)
39.0x 16QAM 3/4	4 (39.0)	12 (78.0)	20 (117.0)	28 (156.0)
52.0x 64QAM 2/3	5 (52.0)	13 (104.0)	21 (156.0)	29 (208.0)
58.5x 64QAM 3/4	6 (58.5)	14 (117.0)	22 (175.5)	30 (234.0)
65.0x 64QAM 5/6	7 (65.0)	15 (130.0)	23 (195.0)	31 (260.0)

II. RELATED WORK

A. *Link Adaptation for SISO WLANs*

Several link/rate adaptation algorithms for the 802.11a/b/g WLANs have been proposed in the literature [6] [7] [8] [9]. Based on the CSI (Channel State Information) used for channel quality estimation, they can be roughly divided into two categories: statistic-based schemes (e.g., ARF [10], Dynamic *ST* [11], AARF [7], SampleRate [9]) and signal-measurement-based schemes (e.g., RBAR [12], Goodput analysis [13], RSS measurement [14]).

B. *Closed-Loop Link/rate Adaptation for MIMO WLANs*

Most of the existing work assumes the closed-loop operation. In [4], the authors proposed a joint PHY and MAC strategy. The PHY scheme maximizes the data rate for a target BER, given a MIMO channel instance. It selects a subset of the total number of transmit antennas and chooses the best constellation that can be supported on each of the selected antennas. The selected rate setting is then fed back from the receiver to the transmitter in the MAC design. Though this protocol is compatible with 802.11a/g, it may not function when communicating with legacy devices.

In [15], the proposed approach evaluates the link quality based on SNR and spatial selectivity information to decide between different diversity and multiplexing modes: beam-forming (BF), double space-time transmit diversity (D-STTD), and spatial multiplexing (SM). The transmission scheme (i.e., BF, D-STTD or SM) that provides the highest throughput for the predefined fixed error rate is selected for a given link. In low SNR case, a beam-forming scheme is selected to increase the robustness of the link. For poor scattering environment and medium SNR, D-STTD scheme can provide additional diversity gain, which results in throughput enhancement. In rich scattering environment and/or high SNR, the adaptive algorithm switches to multiplexing schemes in order to increase spectral efficiency. Though this approach is designed for cellular systems, it can be utilized by other MIMO based systems like 802.11n WLANs.

C. *Open-Loop Link Adaptation for MIMO WLANs*

To the best of our knowledge, we are not aware of any MIMO open-loop algorithms that have been proposed and are publicly available. In [16], the authors designed new combinations of STBC and SDM solutions for the new MCS set of 802.11n, targeting either an increase of peak data rate (SDM) or enhancement of the link (STBC) or a mix of the two using a hybrid approach. It shows that the combinations of open-loop multi-antenna approaches can benefit the system performance a lot, while avoiding the protocol overhead consumed in feedback signalization and calibration process. However, this paper does not specify how to switch among different STBC/SDM schemes or give any further details.

The goal of link/rate adaptation is to maximize throughput and also satisfy any predefined packet error rate (PER) bound, or stabilize PER to some targeted PER. It decides which MIMO mode and MCS should be used, and how long the transmitter should stay at the current transmission mode. This is based on the statistics collected at the sender such as RSSI, ETT (Expected Transmission Time), average retries, PER (Packet Error Rate), delivery ratio and so on.

III. ARFHT OPEN-LOOP LINK ADAPTATION

Controlling the data rate by adjusting the MIMO mode and the modulation and coding scheme can be used to exploit the benefits of independent fading of multi-path propagation. The choice of the transmission scheme has a direct impact on a fundamental property of WLAN, e.g., the throughput. In a WLAN, a static transmission scheme cannot maximize the network's capacity due to the time-varying channel characteristics and rapid change of traffic conditions. It has been shown in [2] that a dynamic transmission scheme is needed to maximize utilization in wireless networks to take advantage of spatial diversity and increased capacity.

As presented in previous sections, there are some basic characteristics of the open-loop (i.e., without perfect channel knowledge at the transmitter) MIMO mode and MCS selection: it uses the same channel coding, the same TX power, and the same MCS for each spatial channel; the feedback consists of binary ACK; there is no sounding, TX beam-forming, calibration, feedback overhead; it offers the ability to spread a single encoded stream across multiple antennas without using closed-loop operation; finally, it is simpler to implement. Such a strategy can be formulated from the extension of SISO link/rate adaptation scheme, while taking account of the new capabilities of 802.11n MIMO PHY. This open-loop extension is particularly important in heterogeneous 802.11 WLANs, for maintaining seamless interoperability and coexistence with legacy devices, which typically, only utilize open-loop link adaptation. We present our ARFHT open-loop link adaptation algorithm in detail in the following sections. Specifically, the objectives of ARFHT are:

(1) Derive a relationship that allows a node to estimate the channel dynamics, based on ACK and the RSSI measured at each RX antenna. Intuitively, this relationship should account for the transmitter's "credits" accumulated in previous transmissions. We are considering maintaining several statistic counters which are dynamically updated.

(2) Determine the probing behavior based on the predicted link quality dynamics: the vertical search regarding the MCS adjustment and the horizontal search for $N_{SS}=N_{STBC}$ adjustment.

(3) Design the open-loop link adaptation rule: basically, this is a threshold-based scheme with the goal of throughput maximizing (or alternatively, minimizing the expected

transmission time). The QoS requirements can also be added by pre-defining the PER and access delay. The objective is to make all parameters adapt to instantaneous channel conditions.

A. Link Quality Estimation

Wireless links are error-prone due to interference, noise, fading, mobility and so on. One of the design goals of ARFHT is to provide efficient link quality estimation in the presence of random wireless errors. In practice, modern devices typically support "multi-rate retry", which retransmits the lost packet at possibly different rates until a success or exceeding the retry limit, to resolve short-lived bursty errors. ARFHT also includes multi-rate retry based error recovery and distinguishes between different types of ACKs. A "partial ACK" is an ACK received after retransmissions of some lost packet. It is a good indication of multiple bursty losses. A "complete ACK" is an ACK received immediately after the first attempt of a packet transmission. Whenever finishing the transmissions of a packet, the transmission status is reported to the upper layer (MAC layer). Then the sender invokes the Link Quality Estimation (LQE) algorithm, which updates the node's various counters of transmission history. Similar to the ARF algorithm in the SISO case, if a row of the partial ACKs are received, LQE classifies the wireless losses to be sustained, which means the channel is deteriorating, and downscale adaptation may be performed. On the other hand, when continuous complete ACKs are received, LQE classifies the wireless link to be in good state or improving, and upscale adaptation may be carried out. Normal transmission/no link adaptation behavior is followed if LQE indicates stable link qualities. The LQE algorithm is shown in Algorithm 1.

Algorithm 1. Link Quality Estimation Algorithm

```
if receiving an ACK then
    update ETT[ ] at the rates used for tx last packet;
    update average RSSIs of recent ACKs measured on RX antennas RSSI[ ];
    timer + +; error = 0; succ[ratecur] + +;
    if receiving a "complete ACK" then
        success + +; failure = 0;
        failureV = 0; failureH = 0;
        successV + = $F_{sv}$(perfect tx(rate), last tx(rate));
        successH+ = $F_{sh}$(minRSSI, maxRSSI);
    else if receiving a "partial ACK" then
        success = 0; failure + +;
        successV = 0; successH = 0;
        failureV + = $F_{fv}$(perfect tx(rate), last tx(rate));
        failureH+ = $F_{fh}$(minRSSI, maxRSSI);
    end if
else if missing an ACK but tries < retry limit then
    fail[ratecur] + +;
    ratecur = Lookup(multirate retry);
else if missing an ACK and tries >= retry limit then
    timer + +; failure = 0; success = 0;
    error + +; err[rate] + +;
end if
```

B. Link Probing

As long as the output of the LQE indicates changing channel qualities, the Link Probing (LP) algorithm is invoked to decide a MCS that may improve the performance. In the SISO case, this decision is simply upscale or downscale by one. However, in the MIMO case, it is now a two-dimensional search process: the search for an appropriate MCS for all spatial streams, and a MIMO mode leveraging spatial diversity and/or spatial multiplexing: the vertical search regarding the MCS adjustment and the horizontal search for N_{SS}/N_{STBC} adjustment (refer to Table 1).

There are 3 MIMO modes: SDM, hybrid STBC/SDM, and STBC. Optional robust transmission rates are achieved when $N_{STS} > N_{SS}$: N_{SS} spatial streams are mapped to N_{STS} space time streams, which are mapped to N_{TX} transmit chains; based either on STBC or hybrid STBC/SDM schemes. The default configuration is no STBC and $N_{STS} = N_{SS}$; for basic MCS set, all spatial streams are encoded with the same MCS; for optional txBF when channel knowledge is available, spatial streams can use different MCS. We only consider the basic ones shown in Table 1 in our simulations.

Several MIMO mode and MCS combinations are possible. The main consideration is to make the MCS adjustment achieve a good balance of sensitiveness (more adaptive) and conservativeness (more safe). The detailed probing process and the MCS update are described in Algorithm 2.

Algorithm 2. Link Probing Algorithm

```
mayUpProbe() :
if (rate + 1)%8! = 0 then
    if successV >= STV then
        newrate = rate + 1;
        if !err[newrate] then
            if isenough(newrate) && fail[newrate] * 2 <= succ[newrate] && ETT[newrate] <= ETT[rate] * timePercentS then
                upProbe = 1;
            else if !isenough(newrate) && perfectETT[newrate] <= ETT[rate] * timePercentS then
                upProbe = 1;
                else if successV >= maxSTV then
                    upProbe = 1;
                end if
            end if
        end if
end if

mayDownProbe();

mayLeftProbe();

mayRrightProbe();

mayLeftUpProbe();

mayRightDownProbe();

nextProbe() :
calculate all possible link probing directions by calling the may[·]Probe() functions;
select the probing direction that would lead to the minimum data rate difference;
return the MCS difference before and after the rate switch (possible values are probe = +1/-1/+7/-7/+8/-8).

needRecoveryFallback() :
if failure > 0 && recovery && rate increased then
    fallback to the previous rate;
end if
recovery = 0;

rateUpdate() :
update rate, $N_{SS}$ and $N_{STBC}$:
recovery = probe;
rate+ = probe;
if probe = = -8 || -7 then
    $N_{SS}$ --; $N_{STBC}$ ++;
else if probe = = +8 || +7 then
    $N_{SS}$ ++; $N_{STBC}$ --;
end if
```

C. Threshold Update

Several thresholds are maintained at the sender: the success thresholds STV and STH, and the failure thresholds FTV and FTH. STV and FTV are related to the MCS adjustment without changing the N_{SS} and N_{STBC}; on the other hand, STH and FTH are referred to change the MCS together with N_{SS} and N_{STBC}. Their values and update rules are illustrated in **Table 2**. Here we use Adaptive Linear Increase Linear Decrease (ALILD) to make the thresholds settings more meaningful. The update details are introduced in Algorithm 3.

Table 2. Thresholds and update rules

Threshold	Update Interval	Update Rule
STV	[minSTV(8), maxSTV(20)]	ALILD
STH	[minSTH(10), maxSTH(25)]	ALILD
FTV	minFTV(3), maxFTV(5)	N/A
FTH	minFTH(6), maxFTH(8)	N/A
timeout	40	N/A
enough	10	N/A
ssThresh[4]	3, 8, 16, 31 [dB]	N/A

Algorithm 3. Threshold Update Algorithm

```
update STV and STH:
if probe = = +1 || -7 then
    $STV += max(rate\%8, 4)$;
    $STV = min(STV, maxSTV)$;
else if probe = = -1 || +7 then
    $STV -= max(rate\%8, 4)$;
    $STV = max(ST, minSTV)$;
end if
if probe = = +8 || +7 then
        $STH += N_{SS}$; $STH = min(STH, maxSTH)$;
else if probe = = -8 || -7 then
        $STH -= N_{SS}$; $STH = max(STH, minSTH)$;
end if
```

D. ARFHT Algorithm

The ARFHT algorithm is comprised of the above three components, namely, Link Quality Estimation (LQE), Link Probing (LP) and Threshold Update (THU).

Algorithm 4. ARFHT Algorithm

1: call LQE to update the statistic counters after each packet transmission;
2: call LP to decide whether or not to probe a new MCS, or need recovery fallback if there is a failure immediately after a MCS adaptation;
3: if the MCS is adapted, update the success thresholds by THU;

IV. SIMULATION METHODOLOGY

A. Building Modules

To the best of our knowledge, no network simulator has a built-in 802.11n PHY model, as the standard itself is very new (an in progress). Therefore, we built our own abstract PHY model and incorporate the PHY layer data into the network simulator (i.e., ns-2). Several modules are used to build the PHY model:

(1) MATLAB TGn Channel Models [17]. It contains the MATLAB scripts to generate the MIMO channel matrix H for Channel Model 'A' to 'F' [18] [19]. There are two main parts: the first consists of scripts to compute a set of correlation matrices for ULAs (Uniform Linear Array); the other can be used to embed the generated MIMO channel into a broader, link-level simulation. This is what we utilize in our simulation.

(2) ns-2 802.11 support [20]. This package develops a new 802.11 module for ns-2 with support for: ET/SNRT/BER-based PHY models, 802.11a multirate, and 802.11e HCCA and EDCA. It also contains an implementation for the ARF [10] and the AARF [7] rate control algorithms. Our implementation of ARFHT is within the same framework of these two algorithms.

(3) Packet error probability prediction model for MIMO-OFDM WLAN systems [21]. This method uses post-detection SNRs as an abstraction of the PHY layer which is sufficient for generating error processes in the system simulations that can accurately reflect the interaction between the MIMO-OFDM PHY layer and the underlying wireless channel. In our simulation, we adopt this abstract PHY model (with some slight modifications) to predict the PERs for Spatial Spreading MIMO processing.

Table 3. PHY Model Parameters

PHY Model Parameters	Value
Carrier frequency	5.25 [GHz]
Signal Bandwidth	20.0 [MHz]
Data subcarriers	52
Tracking pilots	4
GI	4 [usec]
Sampling Rate	80 MHz in 20 MHz mode
Receiver Type	MMSE
MIMO processing	Spatial Spreading
Multi-path fading	Enabled
Shadow fading	Disabled
Transmit Power	50.0 [mW]
Noise Figure	10.0 [dB]
Antenna Gain (TX/RX)	0.0 [dBi]
Antenna Pattern (TX/RX)	Omni-Directional

B. Model Implementation

We built an abstract PHY model (i.e., PER vs. SNR), by including the results of the TGn MIMO Channel Models [17], into the MAC simulator (ns-2-80211 [20]). This is accomplished by modifying the existing modules in Section VI-A as well as adding our own modules. The detailed steps are as follows:

(1) We first configure the MATLAB scripts for TGn Channel Models to generate the various realizations of the channel matrix H, sampled at specified time intervals. Based on H, we implement our own MATLAB scripts for calculation of the RX power of each spatial stream, and the RX SNR measured on each RX antenna, based on the parameters for 802.11n WLAN. Table 3 summarizes these PHY model parameters used. Then the RX power and the RX SNR results are incorporated into the MAC (ns-2-80211) simulator. Specifically, the RX power of each spatial stream is used to calculate the post-detection SNR, and the RX SNR at each RX antenna is used as the RSSI reported by the PHY layer to the MAC layer.

(2) In the ns-2-80211 simulator, a PHY-BER receive model is implemented. Specifically, using the current RX power, the SNIR is calculated against the interference power and the noise power. The SNIR, together with the current data rate and the packet size, are used to calculate the BER, and then the PER. The steps to receive packets in the SISO case are [20]:

- If the SNIR of the first bit of the arriving packet is higher than rxThreshold, lock the reception onto this packet.

- When the last bit of a locked packet is received, parse the event history to reconstruct the piecewise SNIR function during reception of this packet.

- Use the piecewise SNIR function to calculate the probability of receiving correctly each chunk of data over which the SNIR is constant and the transmission mode is the same. Specifically, this is done with $P_i = (1 - BER_i)$^$nbits_i$, where $nbits_i$ is the number of bits over which the SNIR and the transmission mode is constant, BER_i is the BER calculated from a constant SNIR. Of course, it depends on the current transmission mode.

- Calculate the $PER = 1 - \prod_i P_i$

- Draw a random number in a uniform distribution between 0 and 1.

- If the random number is higher than PER, the packet is received correctly. Otherwise, it is received with errors.

- The receiver sends back the ACK if the packet is received correctly.

- Finally, the transmission errors indicated by ACKs are observed by the transmitter (ARFHT MAC station), which performs the link adaptation.

Under the MIMO case, we perform the similar steps listed above, using a sampled RX power read from the RX power file, except that the BER_i and P_i are now calculated for each spatial stream, i.e., BER_{ij} and P_{ij}, where j is the spatial stream index. Then the PER is calculated as $PER = 1 - \prod_j \prod_i P_{ij}$ [21].

(3) At the same time, the ARF MAC station also reads the RSSI reported by the PHY to the MAC for fast link adaptation. Specifically, the PHY layer reports the measured RSSI value from each RX antenna to MAC layer. This parameter is a measure by the PHY sub-layer of the energy observed at the antenna used to receive the current PPDU. RSSI is measured during the reception of the PLCP preamble. It is intended to be used in a relative manner, and is a monotonically increasing function of the received power. The MAC layer may decide MCS based on this information. However, the PHY layer is assumed to provide RSSI value of PCLP header part with inaccuracy. Therefore, we do not use RSSI as the only metric to select the MCS in ARFHT.

(4) New link adaptation algorithm is developed and implemented at the ARFHT MAC station at the MAC layer, as outlined in Section III.

Table 4. Simulation Parameters

Simulation Parameters	Value
Channel model 802.11n	Channel Model B (NLOS)
System configurations	4x4
PPDU length	1000 [bytes]
Target PER	10%
Throughput	TP = (1 - PER)*dataRate
Simulation length	>=4 [s]

C. Simulation Results

In this subsection we provide simulation results for basic MIMO with the configurations shown in Table 4. To validate our PHY model, we first provide the received SNR curves at different distances. For each distance, the simulations were run 50 times, and the RX SNR was measured as an average over all receiver antennas. Fig. 1 shows that our calculated RX SNRs matches well the RX SNRs in TGnSync's simulation results [22]. Later we can use this RX SNR as the RSSI reported by the PHY layer to the MAC layer in the ns-2-80211 simulator.

In the next step, we test the PHY throughput at different distances to validate the PHY-BER receive model. The same settings were used as in the above test. We then compare the obtained PHY throughput with the ones in MIMAC [4], which is a closed-loop approach. Again, Fig. 1 demonstrates that the open-loop ARFHT algorithm can actually provides excellent throughput performance.

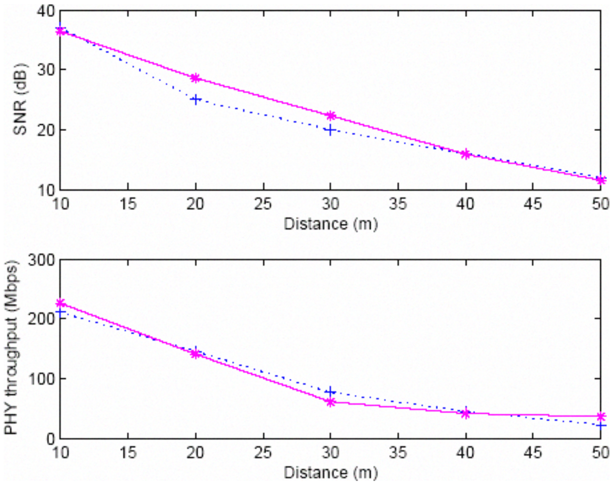

Figure 1. RX SNR / PHY throughput at different distances

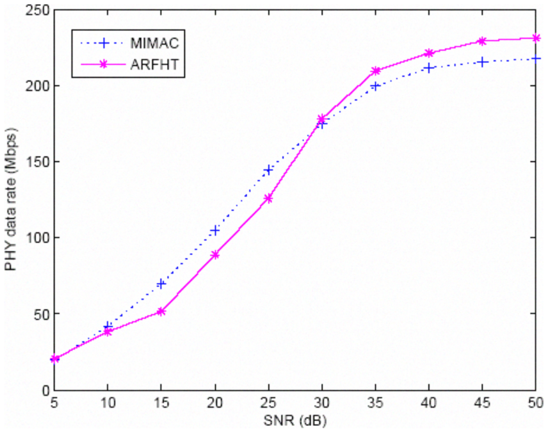

Figure 2. PHY average data rate measured at different post-detection SNRs

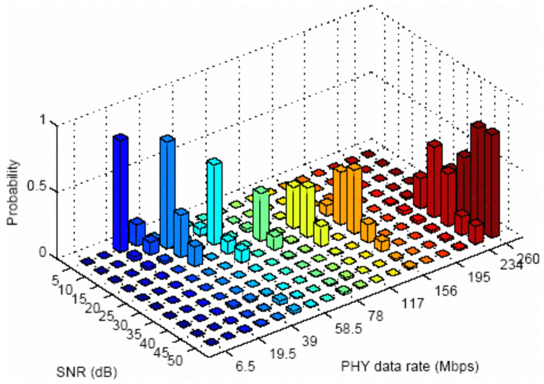

Figure 3. PHY data rate selection probability

Fig. 2 shows the average PHY data rates measured in our simulation as a function of the post-detection SNRs, which are obtained by varying the TX-RX distances. At lower SNRs, ARFHT achieves smaller data rates than MIMAC. This is mainly due to the presence of a lossy link, and ARFHT relies on the arrival of ACKs to adapt the data rate, as well as it requires not exceeding the goal of packet error. As SNR improves, ARFHT can quickly switch to a higher MCS since the ACK arrives continuously. Finally, ARFHT arrives at the highest MCS in the basic MCS set.

Fig. 3 shows the rate selection distribution as a function of post-detection SNRs. We get similar observations like in MIMAC: higher rates can be easily achieved at higher SNRs, since more multiplexing gain is utilized. Also, for a given SNR level, the selected rate is not fixed. Compared to the results obtained in MIMIC, our algorithm distributes the selected rates more evenly. We consider that this is because no exact channel state information is available at the sender; therefore, it is more difficult for the sender to fix a proper MCS (hence the data rate).

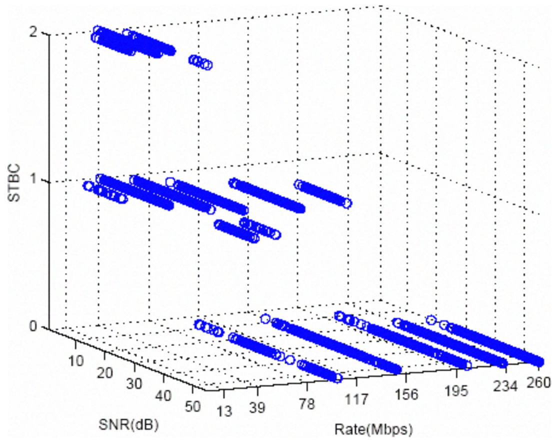

Figure 4. An example of spatial multiplexing and diversity tradeoff

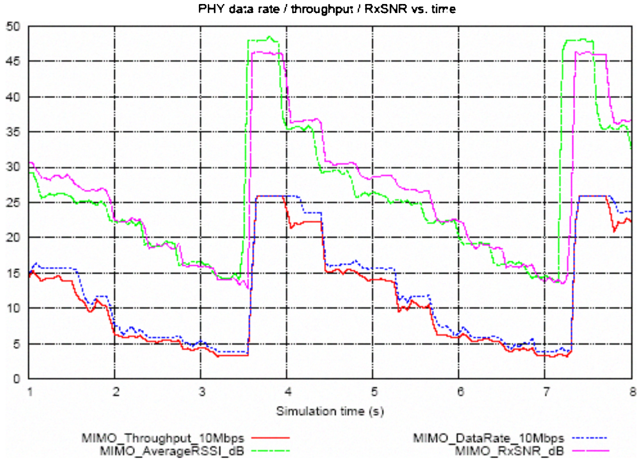

Figure 5. RX SNR / PHY data rate / throughput under decreasing link quality

We further study the effect of multiplexing and diversity tradeoff, shown in Fig. 4. As expected, the open-loop transmit diversity method (i.e., STBC in our case) provides diversity gains (lower BER) at lower SNRs. When SNR improves, multiplexing provides higher throughput gains with more spatial streams.

Fig. 5 demonstrates the adaptability of the ARFHT algorithm. ARFHT changes the data rates quickly with the variations in link quality. At the same time, ARFHT still maintains a small PER (around 10%). This demonstrates that the performed link/rate adaptation is efficient and highly responsive.

V. CONCLUSION

Link/rate adaptation algorithms are extremely important for WLANs with multirate capabilities. Previously proposed solutions are either for SISO WLANs, or closed-loop methods for MIMO WLANs. In this paper, we have proposed and evaluated an open-loop algorithm, ARFHT, which extends the legacy ARF scheme with novel link estimation and probing methods suitable for MIMO WLANs. It maintains the advantages of requiring no changes to the IEEE 802.11 standards and requiring little protocol overhead. ARFHT can adapt to a variety of link conditions, and can be easily adopted for future wireless hardware based on the emerging 802.11n standard. Our next immediate task is to implement this rate/link adaptation scheme on an 802.11n WLAN card under a Linux environment.

REFERENCES

[1] IEEE 802.11 WG, “Draft PAR for High Throughput Study Group,” May 2002.
[2] Enhanced Wireless Consortium, “HT PHY Specification v1.27,” Dec. 2005. http://www.enhancedwirelessconsortium.org/home
[3] Enhanced Wireless Consortium, “HT MAC Specification v1.24,” Jan. 2006. http://www.enhancedwirelessconsortium.org/home
[4] G. Kulkarni, A. Nandan, M. Gerla, and M. Srivastava, “MIMAC: A Rate Adaptive MAC Protocol for MIMO-based Wireless Networks,” in UCLA Technical Report 040035, Dec. 2004.
[5] B. Braswell and J. McEachen, “Modeling Data Rate Agility in the IEEE 802.11a WLAN Protocol,” in Proc. OPNETWORK’01, Mar. 2001.
[6] I. Haratcherev, K. Langendoen, and R. L. H. Sips, “Hybrid Rate Control for IEEE 802.11,” in Proc. MobiWac’04, Philadelphia, PA, Oct. 2004.
[7] M. Lacage, M. Manshaei, and T. Turletti, “IEEE 802.11 Rate Adaptation: A Practical Approach,” in Proc. MSWiM’04, Venice, Oct. 2004.
[8] M. Manshaei, T. Turletti, and M. M. Krunz, “Media-Oriented Transmission Mode Selection in 802.11 Wireless LANs,” in Proc. WCNC’04, Atlanta, Georgia, Mar. 2004.
[9] J. C. Bicket, “Bit-rate Selection in Wireless Networks,” Master’s thesis, Feb. 2005.
[10] A. van der Vegt, “Auto Rate Fallback Algorithm for the IEEE 802.11a Standard,” Utrecht University Tech. Rep., 2002.
[11] P. Chevillat, J. Jelitto, A. N. Barreto, and H. L. Truong, “A Dynamic Link Adaptation Algorithm for IEEE 802.11a Wireless LANs,” in Proc. IEEE ICC, May 2003, pp. 1141–1145.
[12] G. Holland, N. Vaidya, and P. Bahl, “A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks,” in Proc. ACM MOBICOM’01, Rome, Italy, July 2001, pp. 236–251.
[13] D. Qiao, S. Choi, and K. G. Shin, “Goodput Analysis and Link Adaptation for IEEE 802.11a Wireless LANs,” IEEE Trans. on Mobile Computing, vol. 1, no. 4, October-December 2002.
[14] J. del Prado Pavon and S. Choi, “Link Adaptation Strategy for IEEE 802.11 WLAN via Received Signal Strength Measurement,” in Proc. IEEE IICC, vol. 2, Anchorage, Alaska, May 2003, pp. 1108–1113.
[15] A. Forenza, M. Airy, M. Kountouris, R. W. Heath, D. Gesbert, and S. Shakkottai, “Performance of the MIMO Downlink Channel with Multi-Mode Adaptation and Scheduling,” in Proc. SPAWC’05, 2005.
[16] S. Rouquette-L´eveil, A. R. Dias, M. Muck, *et al*, “MIMO-based PHY layer Techniques for IEEE802.11n,” in Wireless World Research Forum 12th meeting, Toronto, Canada, Nov. 2004.
[17] L. Schumacher, “WLAN MIMO Channel Matlab program.” http://www.info.fundp.ac.be/lsc/Research/IEEE 80211 HTSG CMSC/
[18] IEEE 802.11WG, “802.11 TGn Channel Models,” May 2004.
[19] Channel Model Special Committee, “Description of a MATLAB implementation of the Indoor MIMO WLAN channel model proposed by the IEEE 802.11 TGn,” May 2004.
[20] M. Lacage, “ns-2 802.11 support,” Sept. 2005. http://yans.inria.fr/ns-2-80211/
[21] B. A. Bjerke, J. Ketchum, R. Walton, *et al*, “Packet Error Probability Prediction for System Level Simulations of MIMO-OFDM Based 802.11n WLANs,” in Proc. ICC’05, vol. 4.
[22] IEEE 802.11WG, “11-04-0891-06-000n-tgnsync-proposal-phy-results.”