△SNR-MAC: A priority-based multi-round contention scheme for MU-MIMO WLANs

Changwon Nam, Saewoong Bahk∗

Institute of New Media and Communications, Department of Electrical and Computer Engineering, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 151–742, Korea

ARTICLE INFO

Article history:
Received 20 October 2014
Revised 21 July 2015
Accepted 27 August 2015
Available online 9 October 2015

Keywords:
MU-MIMO
MAC protocol
Distributed protocol

ABSTRACT

The performance of uplink multiuser MIMO (MU-MIMO) transmissions heavily depends on which users to transmit together. In WLANs where each user independently determines when to transmit by random access, the performance degradation occurs when a set of users for concurrent transmissions are not chosen properly. In this paper, we address this problem and propose △SNR-MAC protocol to enhance the uplink throughput in MU-MIMO WLANs. In △SNR-MAC, a set of users transmitting together are determined one after another through a multi-round contention where the number of rounds equals the number of antennas at the AP. In each round, given winning users that are already transmitting, each user calculates its SNR reduction amount due to the winning users. △SNR-MAC gives a higher priority to users with less SNR reduction amounts. To achieve this, each round consists of multiple stages where earlier stages are reserved for users with less SNR reduction amounts. In this way, users with the strong channel orthogonality can transmit together in a fully distributed manner. We theoretically analyze the throughput of △SNR-MAC and propose a parameter selection method to maximize the throughput. Our evaluation results confirm that △SNR-MAC improves the uplink throughput over existing schemes both in two- and three-antenna AP cases and achieves temporal fairness in mobile environments.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

With the proliferation of bandwidth-hogging mobile devices such as smartphones and tablet computers, there is an increasing demand for high wireless capacity. Multiple-Input and Multiple-Output (MIMO) has been devised as a promising technology to boost wireless capacity. To handle explosively increasing mobile traffic, most of the wireless systems have embraced the MIMO technology as a solution in their latest standards.

In Wireless Local Area Networks (WLANs), the recent advances have mainly come from the MIMO technology in the physical (PHY) layer. Specifically, the IEEE 802.11n standard has incorporated the single user MIMO (SU-MIMO) with up to four spatial streams, providing a maximum data rate of 600 Mbps [1]. As a next step, the downlink multiuser MIMO (MU-MIMO) has been included in the IEEE 802.11ac standard to further enhance throughput beyond gigabit-per-second [2]. Considering the direction of WLAN evolution, the uplink MU-MIMO is also expected to be available in future standards.

The performance of uplink MU-MIMO transmissions depends on a set of mobile users (STAs) transmitting together. In general, users with the strong channel orthogonality should transmit together to maximize performance. In cellular networks, base users (BSs) can easily control which
users to transmit together through centralized scheduling. In existing user scheduling/grouping algorithms, each BS explicitly selects users for concurrent transmissions and assign their data rates accordingly [3,4]. However, such scheduling algorithms are not suitable for WLANs since random access is used in the medium access control (MAC) layer. In Distributed Coordination Function (DCF) [5], each user independently determines when to transmit while access points (APs) have no scheduling functionality. In this case, the performance degrades severely when users for concurrent transmissions are not chosen well, sometimes to levels worse than that without the MU-MIMO. In order to address this problem, we need a new contention scheme where users with the strong channel orthogonality are selected in a fully distributed manner.

In this paper, we propose ΔSNR-MAC protocol to enhance the uplink throughput in MU-MIMO WLANs. In ΔSNR-MAC, a set of users transmitting together are determined one after another through a multi-round contention where the number of rounds equals the number of antennas at the AP. In each round, given winning users that are already transmitting, each user calculates its SNR reduction amount due to the winning users. ΔSNR-MAC gives a higher priority to users with less SNR reduction amounts. To achieve this, each round consists of multiple stages where earlier stages are reserved for users with less SNR reduction amounts. In this way, users with the strong channel orthogonality can transmit together in a fully distributed manner.

The rest of this paper is organized as follows. The background and motivation are presented in Section 2. We provide the operation procedures of ΔSNR-MAC along with the implementation details in Section 3. The theoretical analysis of throughput is given in Section 4, and the parameter selection method is provided in Section 5. We provide the performance evaluation in Section 6, and an overview of related work in Section 7. We finally conclude this paper in Section 8.

2. Background and motivation

2.1. Uplink multiuser MIMO

In uplink MU-MIMO transmissions, multiple users are enabled to transmit together towards an AP. Fig. 1 shows an example where there are a two-antenna AP and three single-antenna users $u_1$, $u_2$, and $u_3$. When $u_1$ and $u_2$ transmit together, the received signal at the AP is given as

$$
\begin{pmatrix}
  y_1 \\
  y_2
\end{pmatrix} =
\begin{pmatrix}
  h_{11} \\
  h_{21}
\end{pmatrix} \cdot x_1 +
\begin{pmatrix}
  h_{12} \\
  h_{22}
\end{pmatrix} \cdot x_2 +
\begin{pmatrix}
  n_1 \\
  n_2
\end{pmatrix},
$$

where $h\cdot = (h_{ij})$ is the uplink channel vector between $u_i$ and the AP, $x_i$ is the symbol transmitted by $u_i$, and $n_i$ represents the Gaussian noise with variance $N_0$.

The AP decodes the symbols $x_1$ and $x_2$ by using zero-forcing successive interference cancellation (ZF-SIC). To decode a symbol, say $x_2$, first, the AP projects the received signal on a direction orthogonal to $h_1$ as

$$
y_{\text{proj}} = (h_{21}, -h_{11}) \cdot (y_1, y_2)
= (h_{21} n_1 - h_{22} n_2) + (h_{21} n_1 - h_{22} n_2),
$$

where $(\cdot)$ is the inner product. As shown in Fig. 1, this projection allows the AP to null out $x_1$ and decode $x_2$ without interference. The estimation $\hat{x}_2$ of $x_2$ is given by

$$
\hat{x}_2 = x_2 + \frac{h_{21} n_1 - h_{22} n_2}{h_{21} n_1 - h_{22} n_2}.
$$

Since the noise after projection is scaled up, the SNR of $u_2$ decreases due to the projection. The SNR after projection $\text{SNR}_{2,\text{proj}}$ of $u_2$ is given by [7]

$$
\text{SNR}_{2,\text{proj}} = \sin^2(\theta_2) \frac{|h_2 x_2|^2}{N_0} = \sin^2(\theta_2) \text{SNR}_{2,\text{orig}},
$$

where $\theta_2$ is the angle between $h_1$ and $h_2$, and $\text{SNR}_{2,\text{orig}} = |h_2 x_2|^2/N_0$ represents the original SNR when $u_2$ transmits alone.

From (2), we can calculate the SNR reduction amount $\Delta\text{SNR}_2$ (in dB) due to the projection as

$$
\Delta\text{SNR}_2 = 10 \log_{10} \frac{\text{SNR}_{2,\text{orig}}}{\text{SNR}_{2,\text{proj}}} = -20 \log_{10} \sin(\theta_2).
$$

Eq. (3) indicates that the SNR reduction amount depends only on $\theta_2$, and is independent of the original SNR. This is because the channel after projection $h_{2,\text{proj}}$ is determined by $\theta_2$, as shown in Fig. 1. Specifically, there is an inverse relationship between $\theta_2$ and $\Delta\text{SNR}_2$, i.e., the larger $\theta_2$ is, the smaller $\Delta\text{SNR}_2$ is. Also, if $h_1$ and $h_2$ are orthogonal ($\theta_2 = 90^\circ$), the SNR of $u_2$ remains the same even after the projection.

After decoding $x_2$, the AP re-encodes $^1$ and subtract it from the received signal, and decode the other symbol $x_1$. Since there is no projection for $x_1$, the SNR of $u_1$ is simply $\text{SNR}_{1} = |h_1 x_1|^2/N_0$.

The above decoding process can be extended to an $L$-antenna case ($L \geq 3$) with $L$ simultaneously transmitting users.

$^1$ We assume that the AP can estimate $h_2$ perfectly and thus regenerate $h_2 x_2$ without any estimation error.
users. Let $h_i$ denote the channel vector for $u_i$, and $x_i$ denote the symbol transmitted by $u_i$. Suppose that the AP decodes the symbols $x_1, \ldots, x_i$ in order. Let us focus on $x_i$, given that $x_{i-1}, \ldots, x_1$ are successfully decoded and subtracted from the received signal. In order to decode $x_i$, the AP projects the received signal on the direction orthogonal to $h_1, \ldots, h_{i-1}$. In this case, $\theta$ is the angle between $h_i$ and the $(i-1)$-dimensional space spanned by $h_1, \ldots, h_{i-1}$. The value of $\sin \theta$ is given by [7]

$$
\sin \theta = \frac{|h_i \cdot h_i|}{|h_i||h_i|}
$$

where $h_i$ is the vector orthogonal to $S$. Since $S$ is formed by $h_1, \ldots, h_{i-1}$, the SNR reduction amount $\Delta \text{SNR}_i$ of $u_i$ depends only on $h_1, \ldots, h_{i-1}$. That is, the SNR reduction amount of a certain user is determined by the users that will be decoded after it. Also, a user loses more SNR if the AP decodes its symbol earlier, i.e.,

$$
\Delta \text{SNR}_i \geq \cdots \geq \Delta \text{SNR}_1 = 0.
$$

2.2. CCMA and TurboRate

Carrier Counting Multiple Access (CCMA) [6] is an uplink MU-MIMO MAC protocol where users contend for transmission opportunities and join ongoing transmissions one after another. TurboRate [7] is a rate adaptation scheme for the uplink MU-MIMO, operating in conjunction with CCMA. Assuming that ZF is used for decoding, TurboRate enables a user to select an optimal data rate depending on the users that are transmitting together with it.

Fig. 2 shows the operation of CCMA and TurboRate with a three-antenna AP and multiple single-antenna users. All the users contend for the first transmission opportunity using Carrier Sensing Multiple Access (CSMA) as in the 802.11. Once a user, say $u_1$, wins the contention, it starts transmission while the remaining users are aware that a new transmission has started. After transmitting a preamble ($P$), $u_1$ announces its channel vector $h_1$ using the channel information (CH) header. This enables each remaining user to learn $h_1$ that will be used to determine its data rate.

As soon as $u_1$ transmits its payload, $u_2$ and $u_3$ contend for the second transmission opportunity. Since the wireless medium is busy due to $u_1$, CCMA is used for contention. The basic idea behind CCMA is that even though a preamble overlaps with other ongoing transmissions, it is still reliably detectable. That is, the AP and remaining users can detect a preamble sent by another user in the presence of $u_1$’s transmission. This is possible because a preamble has a fixed signal pattern which can be detected using correlation-based mechanisms even in the presence of interference [6].

In Fig. 2, $u_2$ happens to win the contention, transmits its preamble, and waits until $u_1$ pauses its transmission. In TurboRate, winning users stop their ongoing transmissions at a predefined time to allow the next winning user to broadcast its channel vector in a clear medium such that other users can listen to it without interference. Generally, in an $L$-antenna case, there are $(L-2)$ stops for this purpose. When $u_1$ pauses its transmission at $T_{null}$, $u_2$ broadcasts its channel vector $h_2$.

As soon as $u_1$ and $u_3$ resume their transmissions, the remaining users continue contention, and $u_3$ wins the contention. In contrast to $u_2$, $u_3$ does not broadcast its channel vector since no other user will transmit after it. After detecting $u_3$’s preamble, all the remaining users wait until the medium becomes idle again.

In TurboRate, the data rate for each user is selected considering the SNR reduction in the decoding process. Each user estimates its uplink channel by listening to downlink frames, i.e., channel reciprocity, and the AP broadcasts the noise power level. Thus each user can calculate its original SNR. By using ZF for decoding, the AP decodes the received signals in the reverse order of their starting times, e.g., symbols from $u_3$, $u_2$, and $u_1$ in order. Each user calculates its SNR reduction amount and the corresponding after-projection SNR using the channel vectors of the users that won contentions before it. Then it selects its data rate using a mapping table between SNR and data rate. For example, $u_3$ calculates $\Delta \text{SNR}_3$, considering $h_1$ and $h_2$ as in (3), and determines its data rate according to $\text{SNR}_{3, \text{proj}}$. Similarly, $u_2$ calculates $\Delta \text{SNR}_2$ by using $h_1$. In contrast, $u_1$ selects its data rate simply based on its original SNR. In this way, TurboRate enables a user to select its data rate by only considering users that won contentions before it and without knowing users that may transmit after it.

2.3. Motivation and approach

The performance of uplink MU-MIMO transmissions depends on which users to transmit together. We illustrate this by an example shown in Fig. 1. Given that $u_1$ is transmitting, either $u_2$ or $u_3$ can join the concurrent transmission. Since $|h_2|^2$ is equal to $|h_3|^2$, $u_2$ and $u_3$ have the same original SNR. However, from $\theta_3 > \theta_2$, $u_2$ loses more SNR than $u_3$, i.e., $\Delta \text{SNR}_2 > \Delta \text{SNR}_3$, and consequently we have $\text{SNR}_{3, \text{proj}} > \text{SNR}_{2, \text{proj}}$. This means that $u_3$ can use a higher data rate into bits, and included in the CH header. The error of discarding the other taps is negligible. The CH headers for two- and three-antenna cases require 3 and 6 OFDM symbols, respectively.
than $u_2$ and it is more suitable to transmit along with $u_1$. In general, users with the strong channel orthogonality should transmit together to enhance throughput performance.

Considering that random access is used in WLANs, the challenge is how to enable users with the strong channel orthogonality to transmit together in a fully distributed manner. Given some users that are already transmitting, the next transmitting user should have a channel vector orthogonal to those of already transmitting users. To achieve this goal, our approach is to allow only users with small SNR reduction amounts to contend for concurrent transmission. This simple, yet efficient method has the two following benefits:

- We can filter out users which lose more SNR than others and thus potentially degrade the throughput. By minimizing the SNR loss of users that transmit later, we can enhance the throughput substantially.
- We can mitigate collision by excluding users with large SNR reduction amounts from contention. As a result, our approach efficiently copes with a large number of users.

In the following, we present the details of our proposed scheme.

3. $\Delta$SNR-MAC protocol

3.1. Protocol description

At a high level, the proposed $\Delta$SNR-MAC operates as follows:

- Multi-round contention: As in SAM, a set of users transmitting together are determined one after another through a multi-round contention. The number of rounds is equal to the number of antennas at the AP. Also, CSMA/CA is used in round 1 and CCMA is used for subsequent rounds. For efficient contention resolution, each user uses a different backoff counter (BC) for each round, where the backoff counter used in round $i$ is denoted by $BC_i$.
- Priority-based contention participation: While TurboRate allows every user to join contention, $\Delta$SNR-MAC gives a higher priority to users with less SNR reduction amounts. To achieve this, each round consists of multiple stages, and the earlier stages are reserved for users with less SNR reduction amounts.

We now explain the operation of $\Delta$SNR-MAC in an $L$-antenna AP case. In round 1, all the users contend as in the 802.11 by using their BC$1$’s. Once a user wins the contention, it becomes the first winning user, denoted as $w_1$, and announces its channel vector. Then $w_1$ selects its data rate according to its original SNR and adjusts its payload size such that the transmission time of payload is equal to $T_{\text{DATA}}$.

In round $i$, there are $K_i$ stages, each of which contains $M$ SlotTimes. Each user calculates its SNR reduction amount due to $w_1, \ldots, w_{i-1}$, i.e., users that won contentions before round $i$. A user is allowed to contend in stage $k$ if its SNR reduction amount satisfies the following condition:

\[
(k - 1)\delta \leq \Delta \text{SNR} < k\delta, \tag{4}
\]

where $\delta$ (in dB) is the division granularity. A user skips the contention in round $i$ (1) if its after-projection SNR is smaller than the minimum SNR, or (2) its SNR reduction amount is equal to or larger than $K_i\delta$. In this regard, we can consider TurboRate as a special case of $\Delta$SNR-MAC with $K_i = 1$ and $\delta = \infty$.

Each user decides which stage to join according to its SNR reduction amount. If a user is to join stage $k$, it listens to the medium until stage $k$ starts. If it detects a preamble before stage $k$, it knows that some other user has won the contention, and waits until the next round starts. If there is no detected preamble, it joins stage $k$.

For the contention in a stage, each user uses its BC$_i$ which is randomly selected in $[1, M]$. Note that BC$_i$ is newly selected in each round $i$. If a user chooses $m$, then it transmits a preamble at the $(m)$th SlotTime. As a result, the user choosing the smallest BC$_i$ becomes the $i$th winning user, denoted as $w_i$. After transmitting a preamble, $w_i$ waits until round $i$ ends, and announces its channel vector while $w_1, \ldots, w_{i-1}$ pause their transmissions. Also, $w_i$ determines its data rate according to its after-projection SNR and adjusts its payload size such that it ends its transmission at the same time with $w_i$. The other users contending in this stage, after hearing the preamble of $w_i$, wait until the next round starts.

After $w_1, \ldots, w_i$ finish their transmissions, the AP sends an acknowledgment (ACK) to each of them by one by one. According to the transmission result, $w_1$ updates its contention window (CW) and BC$_i$ as in the 802.11. In contrast, $w_2, \ldots, w_i$ simply resume their BC$_i$’s in the next round 1.

We now illustrate an operation example shown in Fig. 3(a). In round 1, $u_1$ starts transmission after winning the contention while the other users freeze their BC$_i$’s and then proceed to round 2. Since there is no user satisfying $\Delta \text{SNR} < \delta$, stage 1 is left idle. In stage 2, $u_2$, $u_3$ and $u_4$ satisfy the condition ($\delta \leq \Delta \text{SNR} < 2\delta$) and join the contention. Since $u_2$ happens to select the smallest number, it wins the contention and transmits its preamble. At the end of round 2, $u_2$ pauses its transmission to allow $u_2$ to broadcast its channel vector. In round 3, $u_3$ joins stage 1 while $u_4$ waits for stage 2. Thus $u_3$ wins the contention and starts its transmission. After hearing the preamble of $u_3$, $u_4$ knows that there are already three ongoing transmissions and waits until the medium becomes idle again. Lastly, the AP sends an ACK to $u_1, \ldots, u_3$ in series. Then $u_1$ randomly generates its new BC$_1$ while $u_2, \ldots, u_4$ resume their BC$_i$’s.

3.2. Collision handling process

If a collision occurs in round 1, the remaining users know from carrier sensing that a new transmission has started. Since colliding users transmit their CH headers at the same time, no user can decode them correctly. Thus, the remaining users skip the subsequent rounds.

If a collision occurs in round $i$ ($i \geq 2$), the remaining users may or may not be able to detect a preamble and behave as follows: (1) if a user (e.g., $u_5$ in Fig. 3(b)) detects a preamble,
it waits until the end of round $i$ where colliding users transmit their CH headers simultaneously. Since the user is unable to decode any of the CH headers correctly, it skips all the subsequent rounds; and (2) if a user (e.g., $u_4$ in Fig. 3(b)) fails to detect a preamble, it continues to participate in contention and transmits in its stage.

Once a collision occurs, the transmissions of colliding users (and users that may transmit later) will fail.

### 3.3. Implementation details

**Channel Estimation and Update:** To estimate the uplink channel, each user leverages channel reciprocity, which is the property that the forward and reverse channels are identical. According to [7], the average difference between an estimation and the actual after-projection SNR is about 0.5 dB, which shows a marginal impact on data rate selection. Another important issue is the channel update period. In $\Delta$SNR-MAC, each user estimates its uplink channel by listening to beacon frames. Generally, the channel update using the beacon frame is sufficient because the indoor channel coherence time ($\geq 0.2$ s) is typically larger than the beacon interval (0.1 s) [7]. However, in fast fading environments, more frequent channel updates are needed. In this case, users that are not the receiver of an ACK frame can opportunistically overhear it for frequent updates.

**Frequency Selectivity:** To deal with frequency selectivity, $\Delta$SNR-MAC uses the effective SNR (ESNR) [11] as TurboRate. Using the per-subcarrier original SNR and its corresponding bit error rate (BER), the average BER over all subcarriers is computed and translated into the original ESNR. Similarly, the after-projection ESNR is calculated using the

---

**Fig. 3.** $\Delta$SNR-MAC operation in a three-antenna AP case. (a) Shows that in round $i$, each user determines which stage to join according to its SNR reduction amount and uses its BC, for contention. (b) Illustrates how users behave in case of collision. In round 2, $u_2$ and $u_3$ transmit their preambles at the same time, resulting in a collision. $u_4$, which has not detected a preamble, continues to participate in contention and transmits in its stage. In contrast, $u_5$, which has detected a preamble, skips subsequent rounds.
after-projection SNR of each subcarrier. Then the SNR reduction amount is given by subtracting the after-projection ESNR from the original ESNR.

Temporal Fairness: Temporal fairness is guaranteed when users share the channel access time fairly [9,10]. ΔSNR-MAC achieves temporal fairness if user channels change independently. This is because users have an equal winning probability with independently changing channels. In addition, the average transmission time of a user, which depends on the stage where it wins the contention, is also the same. As a result, each user has an equal share of channel access time.

Multiple-Antenna Users: We can readily extend ΔSNR-MAC to cover multiple-antenna users. When a user has multiple antennas, each antenna becomes a contending entity with its own backoff counter. If more than one antenna backs off and their BCs reach zero at the same time, only one of them is randomly selected for transmission, and the others perform backoffs again with increased contention window sizes. In this case, a multiple-antenna user can be considered as multiple single-antenna users with virtual collision resolution.

4. Throughput analysis

In this section, we derive the throughput of ΔSNR-MAC analytically in a two-antenna AP case. The key idea is to derive the probabilities of events occurring in each round by considering the distribution of the angle between two channel vectors. The details are given below.

4.1. System model and assumptions

Suppose that there are a two-antenna AP and $N$ single-antenna users. We assume a ring topology where all users are placed on a ring with radius $d$ and the AP is located at the center of the ring. In addition, all users are in the same contention domain without having the hidden terminal problem. Each user always has packets to transmit while there is no downlink traffic. We use the ideal channel model where packet reception errors occur only by collision. The number of stages is denoted by $K$ instead of $K_2$ unless confusion arises.

We assume the i.i.d. Rayleigh flat fading channel where $h_{ij}$ are independent, identically distributed and complex Gaussian random variables. Then the cumulative distribution function (CDF) of the angle $\Theta$ between two independent channel vectors, which is derived in Appendix A, is given as

$$F_{\Theta}(\theta) = 1 - \cos^2 \theta.$$  

(5)

This indicates that, given $w_1$, the CDF of the angle between the channel vectors of a random user and $w_1$ is $F_{\Theta}(\theta)$. In addition, the distribution of original SNR, $\|h\|^2/N_0$, is a chi-squared random variable with the probability density function (PDF) of

$$f_{SNR}(\gamma) = \frac{\gamma^{\frac{\gamma-1}{2}}e^{-\gamma/\text{SNR}_{avg}}}{\text{SNR}_{avg}^{\frac{\gamma+1}{2}}},$$

(6)

where $\text{SNR}_{avg}$ is the average SNR.

4.2. Probabilities for slot types

For analysis, we adopt the technique used in [12–16] with some modifications. We introduce some new definitions of the following. First, we classify packets into two types PKT1 and PKT2. If a packet is transmitted in round 1 (round 2), it is defined as a PKT1 (PKT2). Second, we define a slot as the time interval between two consecutive backoff counter (BC1) decrements [12].

We next classify a slot into five types as follows:

- slot$_1$: There is no transmission attempt.
- slot$_{C2}$: More than one PKT1 is transmitted, resulting in a collision in round 1.
- slot$_{S2}$: One PKT1 is transmitted successfully, but there is no transmission attempt in round 2.
- slot$_{S3}$: One PKT1 and one PKT2 are transmitted successfully.
- slot$_{SC}$: One PKT1 and more than one PKT2 are transmitted, resulting in a collision in round 2.

Notice that successful transmissions occur in slot$_{S2}$ and slot$_{S3}$ while transmissions fail in slot$_{C2}$ and slot$_{SC}$. The probability of each slot type is determined by the probabilities of events occurring in rounds 1 and 2, which are derived as follows.

We first study the behavior of a single user and obtain the probability $p$ that a user transmits in round 1. Let $p$ denote the conditional error probability that a PKT1 is received in error, given that it is transmitted. This happens: (1) when it collides with any other packets or (2) when a collision occurs in round 2. Given that a user transmits a PKT1, case 1 occurs when other users transmit simultaneously. This probability $p_C$ is given by

$$p_C = 1 - (1 - \tau)^{N-1}.$$  

(7)

For case 2, let $p_{C2}$ denote the probability of collision in round 2. Then $p$ can be obtained as

$$p = p_C + (1 - p_C)p_{C2} = 1 - (1 - \tau)^{N-1} + (1 - \tau)^{N-1}p_{C2}.$$  

(8)

where $\tau$ is given as [12]

$$\tau = \frac{2(1 - 2p)}{(1 - 2p)(CW_{\text{min}} + 1) + pCW_{\text{min}}(1 - (2p)^G)}.$$  

(9)

Here $CW_{\text{min}}$ is the minimum contention window size, and $G$ is the maximum backoff stage number. Eqs. (8) and (9) are two non-linear equations with the two unknowns $p$ and $\tau$. Also, $p_{C2}$, which will be derived later, is independent of $\tau$ and $p$. Thus, we can obtain $\tau$ by solving (8) and (9) with numerical techniques [12].

---

6 For a short time period, some users may have strong orthogonality compared to others, and they have higher probabilities to win the contention. However, as long as user channels change independent, such orthogonality happens equally to each user in the long term. As a result, each user wins a contention with an equal probability in the long term.

7 Even though we use the ideal channel model in analysis, we consider the channel fading error in simulation.

---

8 We use the terms “slot” and “SlotTime” for different meanings. A SlotTime refers to a fixed time duration, e.g. 9 μs. In contrast, the duration of a slot varies and is equal to that of a SlotTime only when no transmission occurs in the slot.
By using $r$, we can derive the probability of each event occurring in round 1. Let $P_{1.1}$ denote the probability of round 1 being idle, i.e., no user transmits in round 1. Since $N$ users contend in round 1 and each transmits with the probability $r$, $P_{1.1}$ is given by

$$P_{1.1} = (1 - r)^N. \quad (10)$$

Also, the probability of success in round 1, i.e., only one user transmits in round 1, is

$$P_{3.1} = N\tau(1 - r)^{N-1}. \quad (11)$$

Then the probability of collision in round 1 is simply given as

$$P_{c.1} = 1 - P_{1.1} - P_{3.1}. \quad (12)$$

We now derive the probabilities of events that occur in round 2. Let us first derive the probability $P_{s.2}$ of round 2 being idle. Provided that only one user is transmitting, the remaining $N - 1$ users contend for the second transmission opportunity. If each remaining station has an SNR reduction amount larger than $K$, round 2 becomes idle. Let $\theta_k$ denote the angle when the SNR reduction amount is $x$ (in dB), i.e.

$$x = -20\log_{10}\sin\theta_k.$$

Then $P_{s.2}$ is given by

$$P_{s.2} = F_0(\theta_{K3})^{N-1}, \quad (13)$$

where $F_0$ is the CDF of angle from (5).

Next, we derive the probability $P_{s.2}$ that a successful transmission occurs in round 2. Let $s_k \in (I, S, C)$ be the state of stage $k$, where $I, S$, and $C$ stand for ‘idle’, ‘success’, and ‘collision’, respectively. Then $P_{s.2}$ is given as in (14).

$$P_{s.2} = P(s_1 = S) + P(s_1 = I, s_2 = S) + \cdots + P(s_1 = \cdots = s_{k-1} = I, s_k = S)$$

$$= P(s_1 = S) + \sum_{k=2}^{N} P(s_1 = \cdots = s_{k-1} = I)P(s_k = S|s_1 = \cdots = s_{k-1} = I). \quad (14)$$

where each term in (14) can be obtained as follows.

Let us first derive the probability $P(s_1 = S)$ of success in stage 1. The probability that a user joins stage 1 ($\theta > \theta_\delta$) is

$$P_{join.1} = 1 - F_0(\theta_\delta). \quad (15)$$

Using (15), we obtain the probability that $n$ users join stage 1 as

$$P_{stage.1}(n) = \binom{N-1}{n}(P_{join.1})^n(1 - P_{join.1})^{N-n-1}. \quad (16)$$

Then $P(s_1 = S)$ can be obtained as

$$P(s_1 = S) = \sum_{n=1}^{N-1} P_{stage.1}(n)P_{success}(n, M). \quad (17)$$

where $P_{success}(n, M)$ is the success probability when $n$ users transmit with $M$ SlotTimes.

We next derive the probability of success in stage $k (2 \leq k \leq K)$. In order for stage $k$ to start, stages 1 to $k - 1$ should be idle, i.e., there should be no user with the SNR reduction amount less than $(k - 1)\delta$. This probability is given by

$$P(s_1 = \cdots = s_{k-1} = I) = F_0(\theta_{(k-1)\delta})^{N-1}. \quad (18)$$

Given that stage $k$ starts, the probability that a user joins stage $k$ is

$$P_{join.k} = \frac{F_0(\theta_{(k-1)\delta}) - F_0(\theta_{\delta})}{F_0(\theta_{(k-1)\delta})}. \quad (19)$$

Similar to (16), the probability that $n$ users join stage $k$ is given as

$$P_{stage.k}(n) = \binom{N-1}{n}(P_{join.k})^n(1 - P_{join.k})^{N-n-1}. \quad (20)$$

Then we have

$$P(s_k = S|s_1 = \cdots = s_{k-1} = I) = \sum_{n=1}^{N-1} P_{stage.k}(n)P_{success}(n, M). \quad (21)$$

From (17), (18), and (21), we can calculate Eq. (14).

From $P_{s.1}$ and $P_{s.2}$, the probability $P_{s.2}$ of collision in round 2 is simply given by

$$P_{c.2} = 1 - P_{s.1} - P_{s.2}. \quad (22)$$

Note that $P_{c.2}$ is independent of $r$ as mentioned before.

Finally, we obtain the probability of each slot type as follows:

$$Pr(slot) = \begin{cases} P_{1.1}, & \text{if slot} = \text{slot}_{1i} \\ P_{1.1}, & \text{if slot} = \text{slot}_{c} \\ P_{3.1}P_{s.2}, & \text{if slot} = \text{slot}_{s.1} \\ P_{3.1}P_{c.2}, & \text{if slot} = \text{slot}_{s.2} \\ P_{3.1}P_{c.2}, & \text{if slot} = \text{slot}_{s.3} \end{cases}. \quad (23)$$

4.3. Throughput

Following the definition in [12], we can express the saturation throughput as in (24), where the duration of each slot type is given as follows:

$$s = \frac{E[\text{payload successfully transmitted in a slot}]}{E[\text{length of a slot}]}$$

$$= \frac{Pr(slots_1)E[BS_1] + Pr(slots_2)E[BS_2] + Pr(slots_s)E[BS_s] + Pr(slots_c)E[BS_c] + Pr(slots_{1i})E[BS_{1i}] + Pr(slots_{c2})E[BS_{c2}] + Pr(slots_{cs})E[BS_{cs}] + Pr(slots_{cs})E[BS_{cs}]}{Pr(slots_1) + Pr(slots_2) + Pr(slots_s) + Pr(slots_c) + Pr(slots_{1i}) + Pr(slots_{c2}) + Pr(slots_{cs}) + Pr(slots_{cs})} \quad (24)$$

$$= \begin{cases} T_{i} = T_{SlotTime} \\ T_{C} = T_{S} + T_{P-CH} + T_{DIFS} \\ T_{S} = T_{P-CH} + T_{DATA} + SIFS + T_{ACK} + DIFS \\ T_{SS} = T_{P-CH} + T_{DATA} + 2(SIFS + T_{ACK}) + DIFS \\ T_{P-CH} = T_{P} + T_{CH} \end{cases} \quad (25)$$

where $T_{SlotTime}$, $T_P$, $T_{CH}$, and $T_{ACK}$ are the durations for a SlotTime, a preamble, a channel information, and an ACK packet, respectively. In addition, $E[BS_1]$ and $E[BS_5]$ are the average payload sizes transmitted in slots $s_1$ and $s_{5s}$, respectively. The derivation of $E[BS_1]$ and $E[BS_5]$ is provided in Appendix C.

To validate the analytical model, we compare analysis results with simulation results. The details of simulation setup are explained in Section 6.1. For comparison, we consider 20 stations and change $\delta$ from 0.1 dB to 2.0 dB with the step size of 0.1 dB. Fig. 4 shows the analysis and simulation results for the distances of 20 m, 30 m, and 40 m from the AP. There is a
close match between analysis and simulation with the maximum error rate of less than 5%, confirming the accuracy of our analytical model.

The difficulty in extending the analysis to a general L-antenna case is that the channel vectors of winning users are not independent. In other words, it is not easy to find the distribution of the angle between a channel vector and the subspace spanned by channel vectors of k winning users, which is different from that between a channel vector and the subspace spanned by k independent vectors. However, if we use an empirical CDF obtained through simulations, we can still apply the proposed analytical framework for such cases.

5. Parameter selection

In this section, we describe how the parameters $\delta$ and $M$ are selected to maximize the throughput. We first give some guidelines obtained from the theoretical analysis and then propose a simple table-driven method to select the optimal parameters.

5.1. Guideline from theoretical analysis

In a ring topology, given the number of users $N$ and the distance $d$, the throughput $S$ is determined by $\delta$ and $M$. From (24), we can numerically obtain the throughput for given $M$ and $\delta$, and the result for $N = 20$ and $d = 30\, \text{m}$ is shown in Fig. 5(a). We can see that the throughput is mainly depends on $\delta$, but almost not on $M$\footnote{This is also observed from the results with other values of $N$ and $d$.}. Thus we fix $M = 5$ and only adapt $\delta$ to achieve the maximum throughput.

The throughput $S(\delta)$ for different distances when $N = 10$ is shown in Fig. 5(b). We can observe two important properties as follows:

- $S(\delta)$ is a unimodal function, i.e., it is monotonically increasing for $\delta < \delta^*$ and decreasing for $\delta > \delta^*$. The reason for unimodality can be explained as follows: A smaller $\delta$ leads to less SNR reduction in each stage, but it also incurs a higher stage idle probability. In contrast, a larger $\delta$ prevents each stage from being idle, but results in more collisions and lower data rates. In addition, the throughput of the first transmitting user also decreases with $\delta$ due to more collisions in round 2. As a result, there exits an optimal $\delta^*$ which balances these two opposite effects.
- $\delta^*$ depends only on $N$ but almost not on $d$, e.g., $\delta^* = 0.4$ for every distance. As shown in (3), the SNR reduction amount is independent of the original SNR (which is determined by $d$), and thus $\delta^*$ is insensitive to the network topology. Therefore, we can use a single value of $\delta^*$ for different topologies as long as $N$ is the same.
- The performance gap between $\delta^*$ and a suboptimal $\delta$ is marginal if $\delta$ is close to $\delta^*$. This implies that a small deviation from $\delta^*$ leads to a negligible performance degradation.

5.2. Table-based parameter selection

Since $\delta^*$ is determined by $N$ and is insensitive to the network topology, we can use a simple table-driven method to select $\delta^*$. The basic idea is to pre-establish a look-up table for $\delta^*$ for each value of $N$. As an example, for the considered simple network model, we find $\delta^*$ with the search space of $\{0.05, 0.1, \ldots, 1.5\}$, and the corresponding look-up table is shown in Fig. 6. As expected, we can see that $\delta^*$ decreases with $N$. This is because the impact of smaller $\delta$, i.e., a higher idle probability, is offset by more users.

In general, the i.i.d. Rayleigh fading channel model used in the analysis may not hold in practice, so the look-up table obtained from the simple network model is not valid in realistic environments. Throughout extensive channel measurements, we find that $\delta^*$ for each $N$ is almost the same regardless of network topology and user mobility. As a result, we can construct a look-up table with a sample channel trace set and use it for any other topologies.

In a three-antenna case, along with $M = 5$ and $\delta$, we should determine $K_2$ appropriately. If we increase $K_2$, there will be less transmission time for $w_2$ and $w_3$. Thus we set $K_2 = 1$, i.e., only a single stage in round 2. Throughout simulations, we find that a single stage is still enough for contention resolution with a properly selected $\delta$. In addition, given $K_2$, we can calculate $K_3$ as

$$K_3 = \left[ \frac{T_{\text{DATA}} - K_2 \cdot M \cdot T_{\text{SlotTime}}}{M \cdot T_{\text{SlotTime}}} \right].$$

6. Performance evaluation

6.1. Simulation setup

We evaluate the performance of $\Delta$SNR-MAC by using a link-layer simulator written in MATLAB. In our simulator, we implement all the components of $\Delta$SNR-MAC, assuming that TurboRate and CCMA operate perfectly. We assume a 20 MHz channel in the 5 GHz band where the 802.11n PHY specification is used. To calculate the packet error rate (PER), we adopt the YANS model used in ns-3 with necessary modifications reflecting the 802.11n PHY [17,18]. The SNR range for each data rate is given in Table 1. In addition, the parameters used in simulations are summarized in Table 2.

We compare $\Delta$-SNR MAC with Angle-based Contention Scheme (AngleContention) and TurboRate. AngleContention
allows a user with a small SNR reduction to have a higher probability to win the concurrent transmission opportunity. Specifically, a user selects its contention window size for each contention round according to the angle between its channel and the channels of the ongoing streams. If the angle is large, the user decreases its contention window and earns a higher winning probability. To ensure fairness, if a user decreases (increases) the contention window by \( \delta \) for the current packet, it pays (earns) the priority back by increasing (decreasing) \( \delta \) to the contention window for the next packet.

In simulations, we consider the following two network models:

- Simple network model: We consider a ring topology and i.i.d. Rayleigh flat fading channel. The coherence time is set as 100 ms, and the path loss model B of TGn channel is used, which is for typical office environments with non-line-of-sight (NLOS) conditions [20].
- Trace-driven network model: For channel trace collection, we use Intel Wi-Fi Link (iwl5300) IEEE 802.11 wireless network interface card (NIC) and Linux 802.11n CSI Tool [19]. Intel iwl5300 NIC measures the subcarrier-level SNR for 30 subcarriers, and 802.11n CSI Tool reports the corresponding channel state information (CSI). We collected CSI traces in two different office environments Room-A and Room-B on a building at Seoul National
University. In each room, we consider both static and mobile scenarios, each with 20 CSI traces. For static traces, we located a receiver (AP) at a cubicle and a sender (user) at 20 different cubicles to obtain 20 traces. In addition, we collected mobile traces by walking with the sender at a typical indoor walking speed. We let the sender transmit a packet every 1 ms for 10 s, and each trace contains 10,000 packets.

We use the following notations to explain the simulation results: \( S_j \) is the throughput of \( w_j \); \( S \) is the total throughput, e.g., \( S = S_1 + S_2 \) in a two-antenna case; \( R_i \) is the average data rate of \( w_i \); \( P_{E_j} \) are probabilities of events \( E \in \{ S, C, I \} \) in round \( i \).

### 6.2. Evaluation in simple network model

Fig. 7 shows the performance comparison in a two-antenna AP case for \( N = 30 \) and \( d = 20 \) m. ∆SNR-MAC outperforms TurboRate in \( S, S_1, \) and \( S_2 \), with the gains of 29%, 17%, and 47%, respectively. The throughput gain comes from two factors. First, as shown in Fig. 7(b), ∆SNR-MAC achieves a higher \( R_2 \) (39 Mbps) than TurboRate (30 Mbps) since it gives priority to users with small SNR reduction amounts. Second, ∆SNR-MAC shows a lower \( P_{C,2} \) as in Fig. 7(c) since there are fewer contending users due to the condition for each stage. The gain in \( S_1 \) comes from a lower \( P_{C,2} \) while the combination of lower \( P_{C,2} \) and higher \( R_2 \) contributes to the gain in \( S_2 \).

Fig. 8 shows the performance comparison in a three-antenna AP case for \( N = 30 \) and \( d = 30 \) m. ∆SNR-MAC outperforms TurboRate with a gain of 28.9%. Specifically, the gains in \( S_1, S_2, \) and \( S_3 \) are 9.7%, 26%, and 83%, respectively. We can see a larger gain in \( S_3 \) than in \( S_2 \). Due to ZF, a user loses more SNR as contention rounds continue, e.g., \( w_3 \) loses more SNR than \( w_2 \). As a result, in TurboRate, \( R_3 \) (29 Mbps) is far lower than \( R_2 \) (40 Mbps) with 25% reduction. However, ∆SNR-MAC mitigates this problem and maintains \( R_3 \) (44.9 Mbps) similarly to \( R_2 \) (47.8 Mbps).

We also investigate the impact of parameters \( N \) and \( d \) on throughput gain in a two-antenna AP case. Fig. 9(a) shows the throughput gain when \( N \) ranges from 10 to 30 with \( d = 20 \) m. We can see that the gain increases with \( N \), which is a highly desirable property. In contrast to TurboRate where \( P_{C,2} \) increases with \( N \), ∆SNR-MAC maintains \( P_{C,2} \) almost the same regardless of \( N \) by reducing \( \delta \) as shown in Fig. 6. In addition, \( R_2 \) also increases with \( N \) due to the reduced \( \delta \). Therefore, the throughput gain grows with the number of users.

Fig. 9 (b) shows the gain when \( d = 10, \ldots, 30 \) m with \( N = 30 \). We can see that the gain depends on the distance. When \( d \) is too small, each user has a high original SNR. In this case, even after a user loses its SNR due to projection, it has an after-projection SNR which is still large enough to support the highest data rate of 78 Mbps. As a result, users both in ∆SNR-MAC and TurboRate almost always choose 78 Mbps for \( R_2 \). On the other hand, when \( d \) is too large, each user has

### Table 1
SNR range vs. data rate.

<table>
<thead>
<tr>
<th>SNR (dB)</th>
<th>Data rate (Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1–6.1</td>
<td>6.5</td>
</tr>
<tr>
<td>6.1–8.5</td>
<td>13</td>
</tr>
<tr>
<td>8.5–12.6</td>
<td>19.5</td>
</tr>
<tr>
<td>12.6–15.2</td>
<td>26</td>
</tr>
<tr>
<td>15.2–20.0</td>
<td>39</td>
</tr>
<tr>
<td>20.0–21.2</td>
<td>52</td>
</tr>
<tr>
<td>21.2–22.8</td>
<td>58.5</td>
</tr>
<tr>
<td>22.8–27.0</td>
<td>65</td>
</tr>
<tr>
<td>≥ 27.0</td>
<td>78</td>
</tr>
</tbody>
</table>

### Table 2
Simulation parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SlotTime</td>
<td>9 μs</td>
</tr>
<tr>
<td>SIFS</td>
<td>16 μs</td>
</tr>
<tr>
<td>DIFS</td>
<td>34 μs</td>
</tr>
<tr>
<td>CW_min</td>
<td>31</td>
</tr>
<tr>
<td>CW_max</td>
<td>1023</td>
</tr>
<tr>
<td>TDATA</td>
<td>500 μs</td>
</tr>
<tr>
<td>Ptx</td>
<td>20 dBm</td>
</tr>
<tr>
<td>N0</td>
<td>−93 dBm</td>
</tr>
<tr>
<td>TP</td>
<td>20 μs</td>
</tr>
</tbody>
</table>
a low original SNR which is near the minimum required SNR. Since only users with their after-projection SNR’s larger than the minimum SNR can contend in round 2, contending users even in TurboRate automatically have small SNR reduction amounts.

6.3. Evaluation in trace-driven network model

We conduct trace-driven simulation using the collected channel traces. We consider four different scenarios, i.e. Room-A and Room-B with and without user mobility, where
Fig. 9. Impact of parameters $N$ and $d$ on throughput gain. (a) Shows that the gain increases with $N$ while (b) indicates that the gain is low when $d$ is too small or large.

We now compare the performance of $\Delta$SNR-MAC, Angle-Contention, and TurboRate. We set $\delta^* = 0.5$ for both two- and three-antenna AP cases. Fig. 12 shows that $\Delta$SNR-MAC out-performs TurboRate in every scenario. The throughput gain of $\Delta$SNR-MAC over TurboRate ranges from 18% to 24% in the two-antenna AP case and from 21% to 24% in the three-antenna AP case.

We also investigate temporal fairness by using Jain’s fairness index [21]. Fig. 13 shows the fairness index of $\Delta$SNR-MAC, AngleContention, and TurboRate in four different scenarios. The fairness index of AngleContention and TurboRate is close to 1 in every scenario, meaning that they can guarantee temporal fairness regardless of network topology and user mobility. In contrast, $\Delta$SNR-MAC achieves temporal fairness only in mobile scenarios, and its fairness index is less than 1 in static scenarios. In static scenarios, when some users happen to have strong channel orthogonality, they have higher probabilities to win contentions, leading to an unfairness problem. However, with user mobility, the angle between two channel vectors changes almost independently. As a result, each user has an equal chance of winning contentions, thereby achieving temporal fairness.

7. Related work

There have been extensive researches on the uplink MU-MIMO in the PHY and MAC layers of WLANs. A common
assumption of previous works is that an \( L \)-antenna AP can support up to \( L \) simultaneous transmissions.

Some works deal with MAC protocols that need Request to Send (RTS) / Clear to Send (CTS) exchange to determine which users to transmit together. In [22], a distributed MAC protocol for concurrent uplink transmissions was proposed. If a certain number of RTS frame senders are ready for simultaneous transmissions, whose numbers are no greater than the number of antennas at the AP, all of them are allowed to transmit together and can be confirmed by a CTS frame.

In [23], a superframe structure, which consists of a random access period and a data transmission period, was proposed. In the random access period, users contend for an RTS frame transmission in a multi-round manner until the number of transmitted RTS frames reaches a certain threshold or the duration of the random access period expires. In the data transmission period, users confirmed by a CTS frame transmit data frames simultaneously. A similar approach was proposed in [24], except that orthogonal frequency division multiple access (OFDMA) is used for uplink transmission requests. With the consideration of multi-round contention for RTS transmissions, an optimal contention stopping strategy was proposed in [25], and it is a simple threshold-based rule such that the contention ends as soon as the number of RTS senders exceeds a certain threshold.

The limitations of the above MAC protocols are that: (i) there is a high overhead for the RTS/CTS exchange, (ii) the centralized scheduling is required to determine which users to transmit concurrently, and (iii) multiple orthogonal training sequences are needed for the channel estimation, while there is only one training sequence in the IEEE 802.11 standard.

As a solution to the aforementioned problems, SAM was proposed, which enables multiple users to transmit together without the RTS/CTS exchange [6]. With a decoding technique called chain-decoding, it is possible to decode concurrent transmissions that are asynchronous in time. Moreover, users using CCMA can start uplink transmissions asynchronously, i.e. with different start times, in a fully distributed manner. Based on SAM, TurboRate, which is a rate adaptation scheme for the uplink MU-MIMO, was proposed [7]. It enables a user to select an optimal data rate when transmitting with some other users. In [26], the authors proposed a matching protocol, called MIMO-Mate, that pairs concurrent users in advance. Specifically, the AP first determines which users to transmit concurrently and their transmission order considering both throughput and fairness. Then the AP broadcasts the chain relation in concurrent transmissions so that users can decide when they can transmit based on the predetermined chain relation. In MIMO-Mate, however, the AP needs to calculate and broadcast the matching result, which could incur prohibitive overhead when users move fast or new users join frequently.

Based on SAM and TurboRate, we propose \( \Delta \text{SNR-MAC} \) to improve the uplink throughput by enabling users with closely orthogonal channels to transmit together. In \( \Delta \text{SNR-MAC} \), only users with SNR reduction amounts smaller than a certain threshold are allowed to join the contention. The use of threshold was also proposed in the Multiuser Threshold Selection (Mu-Thres) algorithm [27], which is a downlink MU-MIMO MAC protocol. In the Mu-Thres scheme, after receiving an RTS frame from the AP, only users with SNRs greater than a certain threshold can contend for CTS frame transmissions. Then the AP performs the downlink scheduling according to the channel information feedback through CTS frames. Even though thresholds are used in common for both \( \Delta \text{SNR-MAC} \) and Mu-Thres schemes, there exist some differences: (i) \( \Delta \text{SNR-MAC} \) is for uplink while Mu-Thres scheme for downlink, (ii) there is a high overhead for the RTS/CTS exchange in the Mu-Thres scheme, and (iii)
the centralized downlink scheduling is used in the Mu-thres scheme, while ΔSNR-MAC operates in a fully distributed way.

8. Conclusion

In this paper, we proposed ΔSNR-MAC protocol to improve uplink throughput in MU-MIMO WLANs. In ΔSNR-MAC, simultaneously transmitting users are determined one after another through a multiple-round contention where each user independently decides whether to join the contention or not according to its SNR reduction amount. In this way, users with the strong channel orthogonality can transmit together in a fully distributed manner. We provided the framework for throughput analysis and proposed the parameter selection method to maximize the throughput. Through extensive simulations, we confirmed that ΔSNR-MAC outperforms TurboRate both in two- and three-antenna AP cases and also achieves good long-term temporal fairness.
(a) Two-antenna AP case.

(b) Three-antenna AP case.

Fig. 13. Fairness of $\Delta$SNR-MAC, AngleContention, and TurboRate in four different scenarios.

**Acknowledgment**

This work was supported in part by the National Research Foundation of Korea (NRF) Grant funded by the Korea government (MSIP) (no.2015R1A2A2A01008240) and in part by the MSIP (Ministry of Science, ICT & Future Planning), Korea, under the C-ITRC (Convergence Information Technology Research Center) (IITP-2015-H8601-15-1001) supervised by the IITP (Institute for Information & communications Technology Promotion).

**Appendix A. CDF of angle**

Let $\mathbf{h}_1 = (X + iY, X + iY)$ and $\mathbf{h}_2 = (X + iY, X + iY)$, where $X$ and $Y$ follow the normal distribution $N(0, 1)$. The angle between $\mathbf{h}_1$ and $\mathbf{h}_2$ is defined as $\Theta = \arccos \left( \frac{\mathbf{h}_1 \cdot \mathbf{h}_2}{\|\mathbf{h}_1\| \|\mathbf{h}_2\|} \right)$, which lies in $[0, \pi/2]$. From the definition of $\Theta$, we have $\cos^2(\Theta) = \frac{\|\mathbf{h}_1 \|^2}{\|\mathbf{h}_2\|^2}$, which is uniformly distributed in $[0, 1]$ [28]. Let $Z$ denote a random variable uniformly distributed in $[0, 1]$. Then $\Theta$ can be expressed as $\Theta = \arccos(\sqrt{Z})$. Finally, we can
derive the CDF of θ as follows:

\[ f_\Theta(\theta) = \Pr(\theta \leq \theta) \]

\[ = \Pr(\arccos(\sqrt{Z}) \leq \theta) \]

\[ = \Pr(\sqrt{Z} \geq \cos\theta) \]

\[ = \Pr(Z \geq \cos^2\theta) \]

\[ = 1 - \Pr(Z \leq \cos^2\theta) \]

\[ = 1 - \cos^2\theta. \quad (A.1) \]

In the above, the direction of inequality is reversed in the third step since \(\arccos\) is a decreasing function.

**Appendix B. Success probability in a stage**

Assume that \(n\) stations contend in a stage with \(M\) Slot-Times. It is trivial to obtain the success probability for \(n = 1\) and any \(M\), or \(n > 1\) and \(M = 1\). For \(n > 1\) and \(M > 1\), there should be one station whose \(BC_2\) is smaller than those of all the others. Then we have

\[ P_{\text{success}}(n, M) = \begin{cases} 1, & \text{if } n = 1, \\ 0, & \text{if } n > 1, M = 1, \\ \sum_{m=1}^{M-1} \binom{n}{1} \binom{M-m}{M} n^{-1}, & \text{otherwise.} \end{cases} \quad (B.1) \]

**Appendix C. Average payload size**

We first derive the average data rate \(E[R_1]\) for a PKT1. Suppose that the system can support \(C\) data rates, and let \(D_c\) \((c = 1, \ldots, C)\) denote the \(c\)th data rate. The SNR range is partitioned into \(C\) intervals where \(\Gamma_c = \left[\gamma_c, \gamma_{c+1}\right)\), and the data rate \(DR_c\) is used when the SNR is in \(\Gamma_c\). Using the PDF of SNR in (6), the probability that \(D_c\) is selected is

\[ P(R_1 = D_c) = \int_{\gamma_c}^{\gamma_{c+1}} f_{\text{SNR}}(\gamma) d\gamma. \quad (C.1) \]

Consequently, we have

\[ E[R_1] = \sum_{c=1}^{C} D_c P(R_1 = D_c). \quad (C.2) \]

Next, let us consider the data rate \(R_{2,k}\) for a PKT2 transmitted in stage \(k\). To obtain \(R_{2,k}\), we need the average SNR reduction amount in stage \(k\). Let \(f_\Theta(\theta)\) be the PDF of the angle between two channel vectors, i.e. \(dF_\Theta(\theta)/d\theta\). Also, let \(\Theta^k\) be the angle between the channel vectors of \(w_1\) and \(w_2\) which starts transmission in stage \(k\). The PDF of \(\Theta^k\) is \(f_{\Theta^k}(\theta) = f_{\Theta}(\theta)\) where \(E_k\) represents the event that \(\theta_{k,b} \leq \Theta < \theta_{(k-1)b}\). Then the expectation of \(\Theta^k\) is

\[ E[\Theta^k] = \int_{\theta_{kb}}^{\theta_{(k-1)b}} \frac{\theta f_{\Theta}(\theta)}{F_\Theta(\Theta_{(k-1)b}) - F_\Theta(\Theta_{kb})} d\theta. \quad (C.3) \]

Accordingly, the average SNR reduction amount in stage \(k\) is given as

\[ \Delta\text{SNR}^k = -20 \log_{10}(\sin(E[\Theta^k])). \quad (C.4) \]

Since the data rate for the PKT2 is determined by the after-projection SNR, the probability that \(D_c\) is used is given as

\[ P(R_{2,k} = D_c) = \int_{\Delta\text{SNR}^k + \gamma_c}^{\infty} f_{\text{SNR}}(\gamma) d\gamma. \quad (C.5) \]

Then we have

\[ E[R_{2,k}] = \sum_{c=1}^{C} D_c P(R_{2,k} = D_c). \quad (C.6) \]

We now derive the average payload size \(E[B_{5,1}]\) transmitted in a slots5. Let \(E[B_1]\) be the average payload size of a PKT1 in a slots5. Since the payload duration of the PKT1 is \(T_{\text{DATA}}\), we simply have

\[ E[B_{5,1}] = E[B_1] = E[R_1] T_{\text{DATA}}. \quad (C.7) \]

In the case of a slots5, the average payload size of a PKT1 is simply equal to \(E[B_1]\). However, the average payload size of a PKT2 depends on the stage number in which the transmission starts. The payload duration of a PKT2 transmitted in the \(m\)th SlotTime of stage \(k\) is

\[ T_{k,m} = T_{\text{DATA}} - (k - 1) T_{\text{stage}} - (m - 1) T_{\text{SlotTime}} - T_p. \quad (C.8) \]

where \(T_{\text{stage}}\) is the duration of a stage, i.e. \(MT_{\text{SlotTime}}\). From (C.6) and (C.8), the average payload size of PKT2 transmitted in the \(m\)th timeslot of stage \(k\) is

\[ E[B_{k,m}] = T_{k,m} E[R_{2,k}]. \quad (C.9) \]

Next, let us consider the probability that the transmission of a PKT2 starts at a given time. In a slots5, let \(P_{\text{stage},k}(n|S,S)\) denote the probability that \(n\) stations join the stage \(k\). On condition that the contention in the second round is successfully resolved, we have

\[ P_{\text{stage},k}(n|S,S) = \begin{cases} P_{\text{start},k}(n), & \text{if } k = 1, \\ \sum_{n=1}^{N-1} P_{\text{stage},k}(n|S,S) \left( \binom{n}{1} \right) \left( \frac{M-n}{M} \right)^{n-1}, & \text{otherwise}. \end{cases} \quad (C.10) \]

Then the probability that the transmission of a PKT2 starts in the \(m\)th timeslot of stage \(k\) is

\[ P_{\text{start}}(k, m) = \begin{cases} P_{\text{stage},k}(1|S,S) \left( \binom{1}{1} \right), & \text{if } m = M, \\ \sum_{n=1}^{N-1} P_{\text{stage},k}(n|S,S) \left( \binom{n}{1} \right) \left( \frac{M-n}{M} \right)^{n-1}, & \text{otherwise}. \end{cases} \quad (C.11) \]

where \(\sum_{k=1}^{K} \sum_{m=1}^{M} P_{\text{start}}(k, m) = P_{5,2}\). From (C.9) and (C.11), the average payload size of a PKT2 is given by

\[ E[B_2] = \sum_{k=1}^{K} \sum_{m=1}^{M} E[B_{k,m}] P_{\text{start}}(k, m). \quad (C.12) \]

Finally, we obtain the average payload size \(E[B_{5,5}]\) transmitted in a slots5 as

\[ E[B_{5,5}] = E[B_1] + E[B_2]. \quad (C.13) \]
References


Changwun Nam is currently a Ph.D. candidate in the School of Electrical Engineering and Computer Science, Seoul National University. He received his B.S. degree from Seoul National University, in 2008. His research interests include multi-user mimo networks, full-duplex communications, and radio resource allocation in wireless networks.

Saewoong Bahk received B.S. and M.S. degrees in Electrical Engineering from Seoul National University in 1984 and 1986, respectively, and the Ph.D. degree from the University of Pennsylvania in 1991. From 1991 through 1994, he was with AT&T Bell Laboratories as a Member of Technical Staff where he worked for AT&T network management. In 1994, he joined the school of Electrical Engineering at Seoul National University and currently serves as a Professor. He has been serving as TPC members for various conferences including ICC, GLOBECOM, INFOCOM, PIMRC, WCNC, etc. He is on the editorial boards of IEEE Transaction on Wireless Communications (TWireless), Computer Networks (COMNET), and Journal of Communications and Networks (JCN). His areas of interests include performance analysis of communication networks and network security. He is an IEEE Senior Member and a Member of Whos Who Professional in Science and Engineering.