Optimal Rate Sampling in 802.11 Systems: Theory, Design, and Implementation

Richard Combes, Jungseul Ok, Alexandre Proutiere, Donggyu Yun, and Yung Yi

Abstract—Rate Adaptation (RA) is a fundamental mechanism in 802.11 systems. It allows transmitters to adapt the coding and modulation scheme as well as the MIMO transmission mode to the radio channel conditions, to learn and track the (mode, rate) pair providing the highest throughput. The design of RA mechanisms has been mainly driven by heuristics. In contrast, we rigorously formulate RA as an online stochastic optimization problem. We solve this problem and present G-ORS (Graphical Optimal Rate Sampling), a family of provably optimal (mode, rate) pair adaptation algorithms. Our main result is that G-ORS outperforms state-of-the-art algorithms such as MiRA and Minstrel HT, as demonstrated by experiments on a 802.11n network test-bed. The design of G-ORS is supported by a theoretical analysis, where we study its performance in stationary radio environments where the successful packet transmission probabilities at the various (mode, rate) pairs do not vary over time, and in non-stationary environments where these probabilities evolve. We show that under G-ORS, the throughput loss due to the need to explore sub-optimal (mode, rate) pairs does not depend on the number of available pairs. This is a crucial advantage as evolving 802.11 standards offer an increasingly large number of (mode, rate) pairs. We illustrate the superiority of G-ORS over state-of-the-art algorithms, using both trace-driven simulations and test-bed experiments.

Index Terms—Rate adaptation, multi-armed bandits, 802.11, test-bed

1 Introduction

In wireless communication systems, Rate Adaptation (RA) is a fundamental mechanism allowing transmitters to adapt the coding and modulation scheme to the radio channel conditions. In 802.11 systems, the transmitter may choose from a finite set of rates with the objective of identifying as fast as possible the rate providing maximum throughput, i.e., maximizing the product of the rate and the successful packet transmission probability. The challenge stems from the fact that these probabilities are not known a priori at the transmitter, and that they may evolve over time. The transmitter has to learn and track the best transmission rate, based on the measurements and observations made on the successive packet transmissions. In 802.11 Multiple Input Multiple Output (MIMO) systems (from 802.11n standard), the design of RA mechanisms is further complicated by the fact that the transmitter has to jointly select a rate and a MIMO mode (e.g.,a diversity-oriented single-stream mode or a spatial multiplexing-oriented multiple-stream mode), and hence the best (mode, rate) pair has to be learned and tracked. To simplify the terminology, algorithms that adapt both the rate and MIMO mode are still referred to as RA algorithms.

Over the last decade, a large array of RA mechanisms for 802.11 systems has been proposed. We may categorize these mechanisms depending on the feedback and measurements from past transmissions available at the transmitter, and actually used to sequentially select rates for packet transmissions. Traditionally in 802.11 systems, RA mechanisms are based on rate sampling approaches, i.e., the (mode, rate) selection solely depends on the number of successes and failures of previous packet transmissions at the various available (mode, rate) pairs. Examples of such mechanisms include Auto Rate Fall-back (ARF) [1] and SampleRate [2]. As 802.11 standards evolve, the number of available (mode, rate) pairs increases, and it has become quite large in 802.11n and 802.11ac, making the use of sampling approaches questionable.

A natural alternative to RA sampling approaches consists in using channel measurements. So far, such measurements have not been explicitly used in practice. The most accessible measurement, the Receiver Signal Strength Indicator (RSSI), is known to lead to poor predictions of the Packet Error Rate (PER) at the various rates (see e.g., [3], [4], [5], [6], [7], [8]). These poor predictions are for example due to the fact that RSSI is an indicator averaged over the various Orthogonal Frequency-Division Multiplexing (OFDM) frequency bands, and hence does not reflect frequency-selective fading. Note that 802.11n Network Interface Controllers (NICs) actually measure and report the channel quality at the OFDM sub-carrier level (this indicator is often referred to as the Channel State Information (CSI) in the literature), which provides better information than the simple RSSI. CSI feedback could be used to improve PER prediction accuracy, but incurs extra overhead, see e.g., [3], [4], [7], [9], [10]. CSI feedback is actually supported by very few 802.11 devices. A promising solution could then consist in storing and using only parts of this information, as proposed for example in [8].

- *R. Combes is with Centrale-Supelec, Gif-sur-Yvette 91190, France. E-mail: richard.combes@supelec.fr.*
- *J. Ok is with the KTH Royal Institute of Technology, Stockholm 1428, Sweden, and the Department of Electrical Engineering, KAIST, Daejeon 34141, South Korea. E-mail: jungseul@kth.se.*
- *A. Proutiere is with the KTH Royal Institute of Technology, Stockholm 1428, Sweden, and the INRIA/Microsoft Joint Research Lab, Le Chesnay Cedex 78153, France. E-mail: alepro@kth.se.*
- *D. Yun and Y. Yi are with the Department of Electrical Engineering, KAIST, Daejeon 34141, South Korea. E-mail: dgyun@lanada.kaist.ac.kr, yiyung@kaist.edu.*

Manuscript received 23 Aug. 2017; revised 18 June 2018; accepted 25 June 2018. Date of publication 10 July 2018; date of current version 1 Apr. 2019. (Corresponding author: Richard Combes.)
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TMC.2018.2854758

As of now, it seems difficult to predict whether measurement-based RA mechanisms will be widely adopted in the future, or whether rate sampling approaches will continue to prevail. In this paper, we investigate the fundamental performance limits of sampling-based RA mechanisms. Our objective is to design the *best* possible rate sampling algorithm, i.e., the algorithm that identifies as fast as possible the (mode, rate) pair maximizing throughput. Our approach departs from previous methods to design RA mechanisms: in existing mechanisms, the way sub-optimal (mode, rate) pairs are explored is based on heuristics. In contrast, we look for the optimal way of exploring sub-optimal (mode, rate) pairs. Our main finding is that, perhaps surprisingly, this rigorous approach yields RA mechanisms that outperform state-of-the art algorithms such as MiRA and Minstrel HT, and this claim is supported by experiments on a 802.11n test-bed.

We rigorously formulate the design of the best rate sampling algorithm as an online stochastic optimization problem. In this problem, the objective is to maximize the number of packets successfully sent over a finite time horizon. We show that this problem reduces to a Multi-Armed Bandit (MAB) problem [11]. In MAB problems, a decision maker sequentially selects an action (or an arm), and observes the corresponding reward. Rewards of a given arm are random variables with unknown distribution. The objective is to design sequential action selection strategies that maximize the expected reward over a given time horizon. These strategies have to achieve an optimal trade-off between exploitation (actions that have provided with high rewards so far have to be selected) and exploration (sub-optimal actions have to be chosen so as to learn their average rewards). For our RA problem, the various arms correspond to the decisions available at the transmitter to send packets, i.e., in 802.11a/b/g systems, an arm corresponds to a modulation and coding scheme or equivalently to a transmission rate, whereas in MIMO 802.11n systems, an arm corresponds to a (mode, rate) pair. When a (mode, rate) pair is selected for a packet transmission, the reward is equal to 1 if the transmission is successful, and equal to 0 otherwise. The average successful packet transmission probabilities at the various (mode, rate) pairs are of course unknown, and have to be learnt.

The sequential (mode, rate) selection problem is referred to as a *structured MAB problem* in the following, as it differs from classical MAB problems. Indeed, the average throughputs achieved at various rates exhibit natural structural properties. For 802.11a/b/g systems, the throughput is an *unimodal* function of the selected rate. For MIMO 802.11n systems, the throughput remains unimodal in the rates within a single MIMO mode, and also satisfies some structural properties across modes. We model the throughput as a so-called *graphically unimodal* function of the (mode, rate) pair. As we demonstrate, graphical unimodality is instrumental in the design of RA mechanisms, and can be exploited to learn and track the best rate or (mode, rate) pair quickly and efficiently. Finally, most MAB problems consider stationary environments, which, for our problem, means that the successful packet transmission probabilities at different rates do not vary over time. In practice, the transmitter faces a non-stationary environment as these probabilities could evolve over time. We consider both stationary and non-stationary radio environments.

We present the following contributions.

(i) We formulate the design of optimal RA algorithms as an online stochastic optimization problem, referred to as a *graphically unimodal* Multi-Armed Bandit (MAB) problem (Section 3).

(ii) For stationary radio environments, where the successful packet transmission probabilities using the various (mode, rate) pairs do not evolve, we derive an upper performance bound satisfied by *any* sampling-based RA algorithm. This limit quantifies the inevitable performance loss due to the need to explore sub-optimal (mode, rate) pairs. It also indicates the performance gains that can be achieved by devising RA schemes that optimally exploit the structural properties of the MAB problem. As it turns out, the performance loss due to the need to explore does not depend on the number of available (mode, rate) pairs, i.e., on the size of the decision space. This suggests that rate sampling methods can perform well even if the number of decisions available at the transmitter grows large. We present G-ORS (Graphical-Optimal Rate Sampling) (Section 4), a RA algorithm applicable to 802.11 systems with single or multiple MIMO modes, and whose performance matches the upper bound derived previously. Thus, G-ORS is optimal. As it turns out, the performance of G-ORS does not depend on the size of the decision space (the number of available (mode, rate) pairs), which is quite remarkable, and suggests that sampling-based RA mechanisms perform well even when the decision space is large (Section 5).

(iii) For non-stationary radio environments where the successful packet transmissions do vary over time, we propose SW-G-ORS and EMWA-G-ORS (Section 4) two versions of G-ORS adapted to non-stationary environments, and provide guarantees on the performance of SW-G-ORS. We show that again, the latter does not depend on the size of the decision space, and that the best rate (or (mode, rate) pair) can be efficiently learnt and tracked (Section 6).

(iv) We illustrate the efficiency of our algorithms through simulations exploiting both artificially generated traces, and traces extracted from test-beds (Section 7).

(v) Finally, we implement the proposed algorithms in a 802.11n test-bed, based on the open source Minstrel HT modules. Again we verify that our algorithms outperform state-of-the-art RA schemes, and work well in network scenarios (Section 8). We made the drivers source code publicly available at [12].

2 Related Work

In recent years, there has been a growing interest in the design of RA mechanisms for 802.11 systems, motivated by the new functionalities (e.g., MIMO, and channel width adaptation) offered by the evolving standards.

Sampling-Based RA Mechanisms. ARF [1], one of the earliest RA algorithms, consists in changing the transmission rate based on packet loss history: a higher rate is probed after n consecutive successful packet transmissions, and the next available lower rate is used after two consecutive packet losses. In case of stationary radio environments, ARF essentially probe higher

rates too frequently (every 10 packets or so). To address this issue, AARF [13] adapts the threshold n dynamically to the speed at which the radio environment evolves. Among other proposals, SampleRate [2] sequentially selects transmission rates based on estimated throughputs over a sliding window, and has been shown to outperform ARF and its variants. The aforementioned algorithms were initially designed for 802.11 a/b/g systems, and they seem to perform poorly in MIMO 802.11n systems [10]. One of the reasons for this poor performance is the non-monotonic relation between PER and rate in 802.11n MIMO systems, when considering all rate options and ignoring modes. When modes are ignored, the PER does not necessarily increase with the rate. As a consequence, RA mechanisms that ignore modes may get stuck at low rates. To overcome this issue, the authors of [10] propose MiRA, a RA scheme that zigzags between MIMO modes to search for the best (mode, rate) pair. In the design of RAMAS [14], the authors categorize the different types of modulations into modulation-groups, as well as the MIMO modes into what is referred to as *enhancement* groups; the combination of the modulation and enhancement group is mapped back to the set of the modulation and coding schemes. RAMAS then adapts these two groups concurrently. As a final remark, note that in 802.11 systems, packet losses are either due to a mismatch between the rate selection and the channel condition or due to collisions with transmissions of other transmitters. Algorithms such as MiRA [10], H-RCA [15], LD-ARF [16], CARA [17], and RRAA [18] explicitly distinguish between losses and collisions.

It is important to highlight the fact that in all the aforementioned RA algorithms, the way sub-optimal rates (or (mode, rate) pairs) are explored to identify the best rate is based on heuristics. This contrasts with the proposed algorithms, that are designed, using stochastic optimization methods, to learn the best rate for transmission as fast as possible. The way sub-optimal rates are explored under our algorithms is optimal.

Measurement-Based Methods. As mentioned in the introduction, measurement-based RA algorithms could outperform sampling approaches if the measurements (RSSI or CSI) used at the transmitter could be used to accurately predict the PER achieved at the various rates. However this is not always the case, and measurement-based approaches incur an additional overhead by requiring the receiver to send channel-state information back to the transmitter. In fact, sampling and measurement-based approaches have their own advantages and drawbacks. We report here a few measurement-based RA mechanisms.

In RBAR [19] (developed for 802.11 a/b/g systems), Request To Send/Clear To Send (RTS/CTS)-like control packets are used to "probe" the channel. The receiver first computes the best rate based on the SNR measured over an RTS packet and then informs the transmitter about this rate using the next CTS packet. OAR [20] is similar to RBAR, but lets the transmitter send multiple back-to-back packets without repeating contention resolution procedure. CHARM [21] leverages the channel reciprocity to estimate the SNR value instead of exchanging RTS/CTS packets. In 802.11n with MIMO, ARAMIS [8] uses in addition to the SNR, a metric referred to as *diff*SNR to predict the PER at each rate. The *diff*SNR corresponds to the difference between the maximum and minimum SNRs observed on the various antennas at the receiver. ARAMIS exploits the fact that environmental factors (e.g., scattering, positioning) are reflected in the *diff*SNR. Hybrid approaches combining SNR measurements and sampling techniques have also been advocated, see [22]. It is also worth mentioning cross-layer approaches, as in [23], where the Bit Error Rate (BER) is estimated using information provided at the physical layer.

In some sense, measurement-based RA schemes in 802.11 systems try to mimic RA strategies used in cellular networks. However in these networks, more accurate information on channel condition is provided to the base station [24]. Typically, the base station broadcasts a pilot signal, from which each receiver measures the channel conditions. The receiver sends this measurement, referred to as CQI (Channel Quality Indicator), back to the base station. The transmission rate is then determined by selecting the highest CQI value which satisfies the given Block Error Rate (BLER) threshold, e.g., 10 percent in 3G systems. More complex, but also more efficient rate selection mechanisms are proposed in [25], [26]. These schemes predict the throughput more accurately by jointly considering other mechanisms used at the physical layer, such as HARQ (Hybrid Automatic Repeat Request).

Stochastic MAB Problems. In this paper, we map the design of sampling-based RA algorithms to a so-called graphically unimodal MAB problem. The connection between MAB problems and RA algorithms has been mentioned in [27]. However, the authors of [27] do not solve their MAB problem, and present heuristic algorithms only.

There is an extensive literature on MAB problems, see [28] for a survey. Stochastic MAB formalize sequential decision problems where the decision maker has to strike an optimal trade-off between exploitation and exploration. MAB problems have been applied in many disciplines – their first application was in the context of clinical trials [29]. Most existing theoretical results concern *unstructured* MAB problems [30], i.e., problems where the average reward associated with the various arms are not related. For this kind of problems, Lai and Robbins [11] derived an asymptotic lower bound on regret and also designed optimal decision algorithms. The originality of our problem lies in its structure: the average reward is a graphically unimodal function of the decisions (here the (mode, rate) pairs. This structure is an advantage as it may be exploited to learn the best decision faster, but it also brings additional theoretical challenges. When the average rewards are structured, the design of optimal decision algorithms is challenging. Unimodal bandit problems have received little attention so far. In [31], the authors propose various heuristic algorithms, that in turn do not provide good performance. We recently proposed a solution to unimodal bandit problems in [32]: we derived regret lower bounds and algorithms that approach these performance limits. The present paper builds on these results. Observe that the bandit problem corresponding to the design of RA schemes enjoys a unimodal structure, but also has an additional structure. Indeed, when selecting a rate r for transmission, the expected reward (here the throughput) is the product of r and of the successful packet transmission at this rate. This information should be exploited in the design of RA schemes. We extend the results of [32] to account for this additional structure. We also study unimodal bandit problems in non-stationary environments, where the average rewards of the different arms evolve over time. Non-stationary environments have not been extensively studied in the bandit literature. For unstructured problems, the performance of algorithms based on UCB [33] has been analyzed in [34], [35], [36] under the assumption that the average rewards are abruptly changing. Here we consider more realistic scenarios where the average rewards

smoothly evolve over time. To our knowledge, such scenarios have only been considered in [37], [38], however those contributions do not consider unimodality, which is the main structural assumption we wish to leverage.

3 Preliminaries

3.1 Models

We consider a single link (a transmitter-receiver pair). At time 0, the link becomes active and the transmitter has packets to send to the receiver. For each packet, the transmitter has to select a rate (for 802.11 a/b/g/ systems), or a MIMO mode and a rate (for 802.11n MIMO systems). The set of such possible decisions is denoted by $\mathcal{D}$, and is of cardinality D. The set of MIMO modes is $\mathcal{M}$ (for 802.11 a/b/g systems, there is a single available mode) and in mode m, the rate is selected from set $\mathcal{R}_m$. For $d \in \mathcal{D}$, we write $d = (m, k)$ when the mode m is selected along with the kth lowest rate in $\mathcal{R}_m$. Let r_d be the rate selected under decision d. After a packet is sent, the transmitter is informed on whether or not the transmission has been successful. Based on the observed past transmission successes and failures, the transmitter has to make a decision for the next packet transmission. We denote by Π the set of all possible sequential (mode, rate) pair selection schemes. Packets are assumed to be of unit size so that the duration of a packet transmission at rate r is $1/r$.

3.1.1 Channel Models

For the ith packet transmission using (mode, rate) pair d, a binary random variable $X_d(i)$ indicates the success ($X_d(i) = 1$) or failure ($X_d(i) = 0$) of the transmission.

Stationary Radio Environments. In such environments, the success transmission probabilities using the different (mode, rate) pairs do not evolve over time. This arises when the system considered is static (in particular, the transmitter and receiver do not move). Formally, $X_d(i)$, $i = 1, 2, \ldots$, are independent and identically distributed, and we denote by θ_d the success transmission probability under decision d, $\theta_d = \mathbb{E}[X_d(i)]$. Let $\mu_d = r_d\theta_d$. We denote by $d^\star$ the optimal (mode, rate) pair, $d^\star \in \arg\max_{d\in\mathcal{D}}\mu_d$.

Non-Stationary Radio Environments. In practice, channel conditions may be non-stationary, i.e., the success probabilities could evolve over time. In many situations, the evolution over time is rather slow, see e.g., [27]. These slow variations allow us to devise RA algorithms that efficiently track the best (mode, rate) pair for transmission. In the case of non-stationary environments, we denote by $\theta_d(t)$ the success transmission probability under decision d, and by $d^\star(t)$ the optimal (mode, rate) pair at time t.

Unless otherwise specified, we consider stationary radio environments. Non-stationary environments are treated in Section 6.

3.1.2 Structural Properties: Graphical Unimodality

Our problem is to identify as fast as possible the (mode, rate) pair with the highest throughput. To this aim, we leverage a crucial structural property of the problem. In practice, we observe that the throughput versus (mode, rate) pair function has a structure called *graphical unimodality*.

Definition. Graphical unimodality is defined through an undirected graph $G = (\mathcal{D}, E)$, whose vertices correspond to the available decisions ((mode, rate) pairs). When $(d, d') \in E$, we say that the two decisions d and d' are neighbours, and we let $\mathcal{N}(d) = \{d' \in \mathcal{D} : (d, d') \in E\}$ be the set of neighbours of d. Graphical unimodality means that when the optimal decision is $d^\star$, then for any $d \in \mathcal{D}$, there exists a path in G from d to $d^\star$ along which the expected throughput is strictly increasing. In other words there is no *local* maximum in terms of expected throughput except at $d^\star$. Therefore, the maximum of the expected throughput can be found using local search in G. Formally, $\theta \in \mathcal{U}_G$, where $\mathcal{U}_G$ is the set of parameters $\theta \in [0, 1]^D$ such that, if $d^\star = \arg\max_d \mu_d$, for any $d \in \mathcal{D}$, there exists a path $(d_0 = d, d_1, \ldots, d_p = d^\star)$ in G such that for any $i = 1, \ldots, p$, $\mu_{d_i} > \mu_{d_{i-1}}$. For instance, if $\mathcal{D} = \{1, \ldots, |\mathcal{D}|\}$ and G is a line graph, graphical unimodality reduces to classical unimodality i.e., $d \mapsto \mu_d$ is strictly increasing on $\{1, \ldots, d^\star\}$ and strictly decreasing on $\{d^\star, \ldots, |\mathcal{D}|\}$.

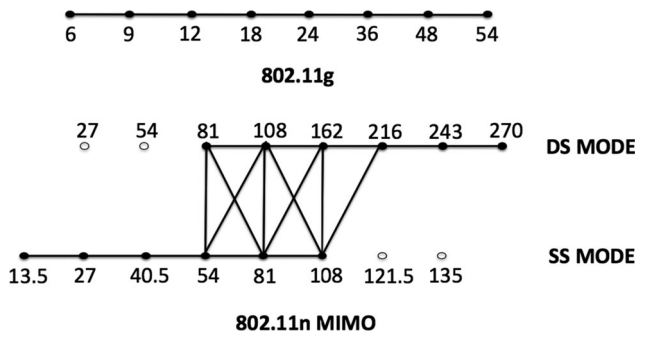


Fig. 1. Graphs G providing unimodality in 802.11g systems (above) and MIMO 802.11n systems (below). Rates are in Mbit/s. In 802.11n, two MIMO modes are considered, single-stream (SS) and double-stream (DS) modes.

Graphical Unimodality in 802.11. In the case of 802.11 systems with a single mode, i.e., 802.11g and earlier standards, the throughput is a unimodal function of the rates, which is well known, see e.g., [10], and hence graphical unimodality holds. The corresponding graph G is a line as illustrated in Fig. 1. In 802.11n MIMO systems, we can find a graph G such that the throughput obtained at various (mode, rate) pairs is graphically unimodal with respect to G. Such a graph is presented in Fig. 1, for systems using two MIMO modes, a single-stream (SS) mode, and a double-stream (DS) mode. It has been constructed exploiting various observations and empirical results from [8], [10]. First, for a given mode (SS or DS), the throughput is unimodal in the rate. Then, when the SNR is relatively low, it has been observed that using SS mode is always better than using DS mode; this explains why for example, the (mode, rate) pair (SS,13.5) has no neighbour in the DS mode. Similarly, when the SNR is very high, then it is always optimal to use DS mode. Finally when the SNR is neither low nor high, there is no mode that clearly outperforms the other, which explains why we need edges between the two modes in the graph.

It is tempting to exploit another structural property of the throughput function. If a transmission is successful at a high rate, it has to be successful at a lower rate, and similarly, if a low-rate transmission fails, then transmitting at a higher rate would also fail. We refer to this property as "monotonicity". Formally this means that for any $m \in \mathcal{M}$, $\theta_{(m,k)} > \theta_{(m,l)}$ if $k < l$, or equivalently that $\theta = (\theta_d, d \in \mathcal{D}) \in \mathcal{T}$, where $\mathcal{T} = \{\eta \in [0, 1]^D : \eta_{(m,k)} > \eta_{(m,l)}, \forall m \in \mathcal{M}, \forall k < l\}$. Unfortunately, this structural property does not always hold [15]. There, it is shown that the rates 9 and 18 Mbps are redundant, and do not satisfy the monotonicity property. Hence we

ignore this property, and focus on exploiting the graphical unimodal structure of the problem.

3.2 Objectives

We now formulate the design of the best (mode, rate) pair selection algorithm as an online stochastic optimization problem. An optimal algorithm maximizes the expected number packets successfully sent over a given time horizon T. The choice of T is not really important as long as during an interval of duration T, a large number of packets can be sent—so that inferring the success transmission probabilities efficiently is possible.

Under a given RA algorithm $\pi \in \Pi$, the number of packets $\gamma^\pi(T)$ successfully sent up to time T is: $\gamma^\pi(T) = \sum_d \sum_{i=1}^{s_d^\pi(T)} X_d(i)$, where $s_d^\pi(T)$ is the number of transmission attempts at (rate, mode) d before time T. The $s_d(T)$'s are random variables (since the rates selected under π depend on the past random successes and failures), and satisfy the following constraint: $\sum_d s_d^\pi(T) \times \frac{1}{r_d} \leq T$. Wald's lemma implies that $\mathbb{E}[\gamma^\pi(T)] = \sum_d \mathbb{E}[s_d^\pi(T)]\theta_d$.

Thus, our objective is to design an online algorithm solving the following stochastic optimization problem:

$$\begin{aligned} &\max_{\pi\in\Pi} \sum_d \mathbb{E}[s_d^\pi(T)]\theta_d, \\ &\text{s.t. } s_d^\pi(T) \in \mathbb{N}, \forall d \in \mathcal{D} \text{ and } \sum_d s_d^\pi(T) \times \frac{1}{r_d} \leq T. \end{aligned} \tag{1}$$

3.3 Graphically Unimodal Multi-Armed Bandit

We now show that problem (1) is asymptotically (for large T) equivalent to a *graphically unimodal* MAB problem. Consider an alternative system where the duration of a packet transmission at any rate is one slot, and where decisions are taken at the beginning of each slot. When rate r is selected, and the transmission is successful, the reward is incremented by an amount of r bits. In this alternative system, the objective is to design $\pi \in \Pi$ solving the following optimization problem.

$$\begin{aligned} &\max_{\pi\in\Pi} \sum_d \mathbb{E}[t_d^\pi(T)]r_d\theta_d, \\ &\text{s.t. } t_d^\pi(T) \in \mathbb{N}, \forall d \in \mathcal{D}, \text{ and } \sum_d t_d^\pi(T) \leq T, \end{aligned} \tag{2}$$

where $t_d^\pi(T)$ denotes the number of times decision d has been taken up to slot T. If the same algorithm π is applied both in the original and alternative systems, we simply have: $t_d^\pi(T) = s_d^\pi(T)/r_d$, assuming without loss of generality that $1/r_d$ is an integer number of slots. The optimization problem (2) corresponds to a MAB problem (see below for a formal definition). To assess the performance of $\pi \in \Pi$, it is usual in the MAB literature to use the notion of *regret*. The regret up to slot T compares the performance of π to that achieved by an Oracle algorithm always selecting the best (mode, rate) pair. The regrets $R_1^\pi(T)$ and $R^\pi(T)$ of algorithm π up to time slot T in the original and alternative systems are then:

$$\begin{aligned} R_1^\pi(T) &= \theta_{d^*} \lfloor r_{d^*} T \rfloor - \sum_d \theta_d \mathbb{E}[s_d^\pi(T)], \\ R^\pi(T) &= \theta_{d^*} r_{d^*} T - \sum_d \theta_d r_d \mathbb{E}[t_d^\pi(T)]. \end{aligned}$$

In the next section, we show that for any $\pi \in \Pi$, an asymptotic lower bound of the regret $R^\pi(T)$ is of the form $c(\theta)\log(T)$ where $c(\theta)$ is a strictly positive constant. It will also be shown that there exists an algorithm $\pi \in \Pi$ that actually achieves this lower bound in the alternative system, in the sense that $\limsup_{T\to\infty} R^\pi(T)/\log(T) \leq c(\theta)$. In such a case, we say that π is asymptotically optimal. The following lemma states that the same lower bound holds in the original system, and that any asymptotically optimal algorithm in the alternative system is also asymptotically optimal in the original system. All proofs are presented in Appendix.

Lemma 3.1. *Let $\pi \in \Pi$. For any $c > 0$, we have:*

$$\begin{aligned} \left(\liminf_{T\to\infty} \frac{R^\pi(T)}{\log(T)} \geq c\right) &\Longrightarrow \left(\liminf_{T\to\infty} \frac{R_1^\pi(T)}{\log(T)} \geq c\right), \\ \left(\limsup_{T\to\infty} \frac{R^\pi(T)}{\log(T)} \leq c\right) &\Longrightarrow \left(\limsup_{T\to\infty} \frac{R_1^\pi(T)}{\log(T)} \leq c\right). \end{aligned}$$

In view of the above lemma, instead of trying to solve (1), we can rather focus on analyzing the MAB problem (2). We know that optimal algorithms for (2) will also be optimal for the original problem. Our MAB problem, whose specificity lies in its structure, i.e., in the correlations and graphical unimodality of the throughputs obtained using different (mode, rate) pairs, is summarised below.

(P_G) *Graphically Unimodal MAB.* We have a set $\mathcal{D}$ of possible decisions. If decision d is taken for the ith time, we receive a reward $r_d X_d(i)$. $(X_d(i), i = 1, 2, \ldots)$ are i.i.d. with Bernoulli distribution with mean θ_d. The structure of rewards across decisions are expressed through $\theta \in \mathcal{U}_G$ for some graph G. The objective is to design an algorithm π minimizing the regret $R^\pi(T)$ over all possible algorithms $\pi \in \Pi$.

4 G-ORS Algorithms

We propose G-ORS (Graphical-Optimal Rate Sampling), a family of (rate, mode) selection algorithms which can be adapted to both stationary and non-stationary environments.

4.1 Algorithm Description

G-ORS algorithms use three statistics in order to select a decision at time slot n: $t_d(n)$ which is the number of times decision d has been selected under G-ORS up to slot n, $\hat{\mu}_d(n)$ the empirical average reward using decision d up to slot n. More precisely $\hat{\mu}_d(n)$ equals r_d times the empirical success probability using decision d up to slot n. The leader $L(n)$ at slot n is the decision with maximum empirical average reward $\hat{\mu}_d(n)$ (ties broken arbitrarily). The last statistic is $l_d(n)$, the number of times decision d has been the leader up to slot n. The precise definitions of $t_d(n)$, $\hat{\mu}_d(n)$ and $l_d(n)$ are given in the next sections, and allow to adapt the algorithm to both stationary and non-stationary environments. Introduce, for any $d \in \mathcal{D}$, the set $M(d) = \mathcal{N}(d) \cup \{d\}$. Finally, let γ be the maximum degree of a vertex in G. The G-ORS algorithm assigns an index to each decision d and the index $b_d(n)$ of decision d in time slot n is given by $b_d(n) = F(t_d(n), \hat{\mu}_d(n), l_{L(n)}(n), r_d)$ where:

$$F(t, \mu, l, r) = \max\left\{q \in [0, r] : tI\left(\frac{\mu}{r}, \frac{q}{r}\right) \leq \log l + c\log\log l\right\}, \tag{3}$$

with $c \geq 3$ a positive constant, and I the Kullback-Leibler (KL) divergence between two Bernoulli distributions with respective means p and q:

$$I(p, q) = p\log\frac{p}{q} + (1-p)\log\frac{1-p}{1-q}.$$

For the nth slot, G-ORS essentially selects the decision in the neighborhood of the leader with maximal index and is described in Algorithm 1. Ties are broken arbitrarily.

Algorithm 1. G-ORS Algorithm

For $n = 1, \ldots, D$:
Select (mode, rate) pair $d(n) = n$.
For $n \geq D+1$:
$L(n) \in \arg\max_{d \in \mathcal{D}} \hat{\mu}_d(n)$;
$\bar{d} \in \arg\max_{d \in M(L(n))} b_d(n)$;
Select (mode, rate) pair

$$d(n) = \begin{cases} L(n) & \text{if } (l_{L(n)}(n)-1)/\gamma \in \mathbb{N}, \\ \bar{d} & \text{otherwise.} \end{cases}$$

4.2 G-ORS Design Principles

The rationale of G-ORS comes from Theorem 5.1, stated in the next section. This theorem quantifies the minimal number of times each sub-optimal decision $d \neq d^\star$ *should* be explored. More precisely, we show in the proof of Theorem 5.1 that d should be explored $\log(T)/I(\theta_d, \mu_{d^\star}/r_d)$ times if d is a neighbor of $d^\star$ in G, and a sub-logarithmic number of times otherwise. Now since $d^\star$ is unknown, the leader $L(n)$ is used as a surrogate. To check if $L(n) = d^\star$, G-ORS selects the neighbor of $L(n)$ with largest index $b_d(n)$. The index $b_d(n)$ is a high probability confidence upper bound for μ_d, so that selecting the decision with highest index ensures that the best neighbor of $L(n)$ will be discovered after exploring for a sufficiently long time. Therefore G-ORS conducts a local search in graph G by exploring neighbors of the current leader, and moves towards decisions which are better than the current leader. In fact, asymptotically, the proportion of time where $L(n) \neq d^\star$ is negligible, so that only neighbors of $d^\star$ incur significant regret (see Theorem 5.2). Also, if G is the complete graph, the problem reduces to a classical MAB, and G-ORS reduces to a variant of KL-UCB, an asymptotically optimal for classical MABs [39].

4.3 Variants

Depending on the definition of the statistics $t_d(n)$, $\hat{\mu}_d(n)$ and $l_d(n)$, we now define several variants of G-ORS. Let $d(t)$ denote the (mode, rate) pair selected at time t under G-ORS.

4.3.1 Basic G-ORS

The basic G-ORS algorithm, which is adequate for stationary environments is obtained by computing the statistics using the complete history from time slot 1 to n, so that:

$$t_d(n) = \sum_{t=0}^{n} \mathbf{1}\{d(t) = d\}$$

$$\hat{\mu}_d(n) = \sum_{t=0}^{n} \frac{r_d}{t_d(n)} X_d(t) \mathbf{1}\{d(t) = d\}$$

$$l_d(n) = \sum_{t=0}^{n} \mathbf{1}\{L(t) = d\}.$$

4.3.2 SW-G-ORS

The SW-G-ORS algorithm (Sliding Window G-ORS) is adequate for non-stationary environments and computes statistics (denoted with a $.^\tau$ superscript) on a sliding window of size τ:

$$t_d^\tau(n) = \sum_{t=n-\tau}^{n} \mathbf{1}\{d(t) = d\}$$

$$\hat{\mu}_d^\tau(n) = \sum_{t=n-\tau}^{n} \frac{r_d}{t_d^\tau(n)} X_d(t) \mathbf{1}\{d(t) = d\}$$

$$l_d^\tau(n) = \sum_{t=n-\tau}^{n} \mathbf{1}\{L^\tau(t) = d\}.$$

Indeed a natural and efficient way of tracking the changes of $\theta(t)$ over time is to select a decision at time t based on observations made over a fixed time window preceding t, i.e., to account for transmissions that occurred between time slots $t-\tau$ and t, see e.g., [36]. The time window τ is chosen empirically: it must be large enough (to accurately estimate throughputs), but small enough so that the channel conditions do not vary significantly during a period of duration τ.

4.3.3 EWMA-SW-ORS

One issue with SW-G-ORS is that it requires us to keep and constantly update in memory the history of all transmissions and their outcomes over the last time window. This can becomes even impossible for large windows. A classical way to circumvent this problem is to use Exponentially Weighted Moving Averages (EWMAs) instead of sliding windows [36]. When using EWMA in the computation of quantities such as $\hat{\mu}_d(n)$, past transmissions are discounted at a rate parametrized by $\alpha > 0$ based on the time epochs they occurred:

$$t_d^\alpha(n) = (1-\alpha) t_d^\alpha(n-1) \; + \alpha \mathbf{1}\{d(t) = d\}$$

$$\hat{\rho}_d^\alpha(n) = (1-\alpha) \hat{\rho}_d^\alpha(n-1) + \alpha r_d X_d(t) \mathbf{1}\{d(t) = d\}$$

$$l_d^\alpha(n) = (1-\alpha) l_d^\alpha(n-1) \; + \alpha \mathbf{1}\{L^\alpha(t) = d\}$$

$$\hat{\mu}_d^\alpha(n) = \hat{\rho}_d^\alpha(n) / t_d^\alpha(n).$$

with the convention that $t_d^\alpha(0)$, $\hat{\rho}_d^\alpha(0)$ and $l_d^\alpha(0)$ are 0. In Section 8, we present another practical way to discount past transmissions so as to get an algorithm with memory requirement as light as that of EWMA-G-ORS, and that mimics the behavior of SW-G-ORS.

5 Stationary Radio Environments

We consider here stationary radio environments, and first derive a lower bound on regret satisfied by *any* (mode, rate) selection algorithm. Then, we show that the regret of G-ORS matches the derived lower bound.

5.1 Regret Lower Bound

To derive a lower bound on regret for MAB problem (P_G), we first introduce the notion of *uniformly good* algorithms [11]. An algorithm π is uniformly good, if for all parameters θ, for any $\alpha > 0$, we have[1]: $\mathbb{E}[t_d^\pi(T)] = o(T^\alpha), \forall d \neq d^*$, where $t_d^\pi(T)$ is the number of times decision d has been chosen up to time slot T, and d^* denotes the optimal decision (d^* depends on θ). Uniformly good algorithms exist as we shall see later on. We further define, for any $d \in \mathcal{D}$, the set $N(d) = \{d' \in \mathcal{N}(d) : \mu_d \leq r_{d'}\}$.

Theorem 5.1. *Let $\pi \in \Pi$ be a uniformly good sequential decision algorithm for the MAB problem (P_G). We have:*

$$\liminf_{T \to \infty} \frac{R^\pi(T)}{\log(T)} \geq c_G(\theta),$$

where $c_G(\theta) = \sum_{d \in N(d^\star)} \frac{r_{d^\star}\theta_{d^\star} - r_d\theta_d}{I(\theta_d, \frac{r_{d^\star}\theta_{d^\star}}{r_d})}$.

1. $f(T) = o(g(T))$ means that $\lim_{T\to\infty} f(T)/g(T) = 0$.

The number of terms in the sum $c_G(\theta)$ is at most equal to the degree of the graph G. In particular, in case of 802.11 systems with a single mode, G is a line, and $c_G(\theta)$ has at most two terms. In MIMO 802.11n systems, $c_G(\theta)$ has at most 4 terms if G is the graph presented in Fig. 1. More generally, the regret lower bound does not depend on the number of available decisions, which is an important property as this number can be quite large. Note that to obtain this lower bound, the graphical unimodality of the throughput plays an important role. Indeed, without structure, the lower bound on regret scales linearly with the number of available decisions, see [11].

5.2 Regret Analysis of G-ORS

Next we show that the regret of G-ORS matches the lower bound derived in Theorem 5.1, i.e., under G-ORS, the way suboptimal (rate, mode) pairs are explored to identify the best pair $d^\star$ is optimal. The next theorem states that the regret achieved under G-ORS matches the lower bound derived in Theorem 5.1. G-ORS is hence asymptotically optimal.

Theorem 5.2. *Fix $\theta \in \mathcal{U}_G$. The regret of algorithm π = G-ORS satisfies:*

$$\limsup_{T\to\infty} \frac{R^\pi(T)}{\log(T)} \leq c_G(\theta).$$

It is worth noting again that the regret of G-ORS does not depend on the size of the decision space, which, as already mentioned, constitutes a highly desirable property. In the proof of the above theorem, we actually get a more precise bound on $R^\pi(T)$: it is shown that for any $\epsilon > 0$, $R^\pi(T) \leq (1+\epsilon)c_G(\theta)\log(T) + O(\log(\log(T)))$.

6 Non-Stationary Radio Environments

In this section, we consider non-stationary radio environments where the transmission success probabilities $\theta(t)$ at various (mode, rate) pairs evolve over time. We show that SW-G-ORS efficiently tracks the best mode and rate for transmission, provided that the speed at which $\theta(t)$ evolves remains controlled. To simplify the presentation, we present the algorithm in the alternative system (see Section 3), where time is slotted, and at the beginning of each slot, a (mode, rate) pair is selected, i.e., we study non-stationary versions of MAB problem (P_G).

6.1 Objective

We denote by $X_d(t)$ the binary r.v. indicating the success or failure of a transmission using (mode, rate) pair d at the tth slot. $(X_d(t), t = 1, 2, \ldots)$ are independent with evolving mean $\theta_d(t) = \mathbb{E}[X_d(t)]$. Let $\mu_d(t) = r_d\theta_d(t)$. The objective is to design a sequential decision scheme minimizing the *non-stationary* regret $R^\pi_{\mathrm{NS}}(T)$ over all possible algorithms $\pi \in \Pi$, where

$$R^\pi_{\mathrm{NS}}(T) = \sum_{t=1}^{T} \Big(\mu_{d^\star(t)}(t) - \mu_{d^\pi(t)}(t)\Big),$$

and $d^\star(t)$ (resp. $d^\pi(t)$) denotes the best decision (resp. the decision selected under π) at time t. $d^\star(t) = \arg\max_d \mu_d(t)$. The above definition of regret is not standard: the regret is exactly equal to 0 only if the best transmission decision is known at each time. This notion of regret really quantifies the ability of the algorithm π to track the best decision. In particular, as shown in [36], under some mild assumptions on the way $\theta(t)$ varies, we cannot expect to obtain a regret that scales sub-linearly with time horizon T. The regret is linear, and what we really wish to minimize is the regret per unit time $R^\pi_{\mathrm{NS}}(T)/T$.

6.2 Assumptions

To analyze the performance of SW-G-ORS, we make the following assumptions. $\theta(t)$ varies smoothly over time. For any d, $\theta_d(t)$ is σ-Lipschitz (i.e., $|\theta_d(t') - \theta_d(t)| \leq \sigma|t' - t|$). We further assume that graphical unimodality holds at all time, in the sense that for any t, $\theta(t) \in \bar{\mathcal{U}}_G$, where $\bar{\mathcal{U}}_G$ is the smallest closed set containing $\mathcal{U}_G$ (taking the closure of $\mathcal{U}_G$ is needed: the optimal decision changes, and hence at some times, two decisions may have the same average throughput). Finally, we assume that the proportion of time where two decisions are not well separated (they have similar throughput) is controlled in the following sense: there exists Δ_0 and $C > 0$ such that for any $\Delta \leq \Delta_0$, for any d and $d' \in N(d)$,

$$\limsup_{T\to\infty} \frac{1}{T}\sum_{n=1}^{T} \mathbf{1}\{|\mu_d(n) - \mu_{d'}(n)| \leq \Delta\} \leq C\Delta.$$

This assumption is natural, and typically holds in practice: the constant C characterizes the proportion of time when the throughputs achieved under d and d' cross each other. In MAB problems, it is in general problematic to have decisions with very similar average rewards, and this constitutes the main difficulty in the regret analysis in non-stationary environments where rewards of various decisions cross each other. The above assumptions are only required to provide a theoretical regret analysis of SW-G-ORS and do not impact its practical applicability as σ, C and Δ are not input parameters of SW-G-ORS (or EMWA-G-ORS).

6.3 Regret Analysis of SW-G-ORS

Theorem 6.1. *Under the above assumptions, the non-stationary regret under π =SW-G-ORS satisfies:*

$$\limsup_{T\to\infty} \frac{R^\pi_{\mathrm{NS}}(T)}{T} \leq C'\sigma^{1/4}\log(1/\sigma),$$

where the constant $C' > 0$ is uniform in σ.

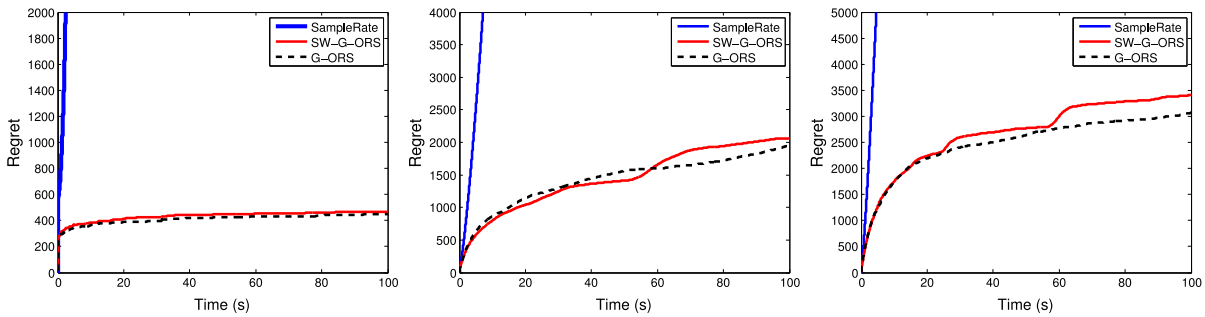
Note that $\sigma^{1/4}\log(1/\sigma)$ tends to 0 as $\sigma \to 0$, which indicates that the regret per unit of time vanishes when we slow down the evolution of $\theta(t)$, i.e., SW-G-ORS tracks the best decision if $\theta(t)$ evolves slowly. Also observe that the performance guarantee on SW-G-ORS does not depend on the size of the decision space (i.e., on D). The proof of the above theorem is long and technical, and is omitted here. It is presented in details in [40], and follows similar techniques as those used in the proof of Theorem 5.1 in [32].

7 Trace-Based Evaluation

In this section, we evaluate the performance of RA algorithms using both artificially generated data and test-bed traces. The test-bed traces contain, at each time, the outcomes of transmissions for the various possible decisions. Hence, this trace-based evaluation allows a fair performance comparison of the different RA algorithms. It further provides the opportunity to assess the performance of an Oracle algorithm always selecting the decision with the highest expected throughput, and thus to quantify the regret of the various algorithms.

7.1 802.11g Systems

We start with 802.11g systems with 8 available rates, namely $\{6, 9, 12, 18, 24, 36, 48, 54\}$ Mbits/s. We test four RA



(a) Steep scenario (b) Gradual scenario (c) Lossy scenario

Fig. 2. Regret in 802.11g systems for SampleRate, G-ORS, and SW-G-ORS in three different stationary scenarios: (a) Steep, (b) gradual, and (c) lossy scenarios. The unit of regret is 0.5 Kbits.

algorithms: G-ORS, SW-G-ORS, SampleRate [2], and Oracle. We use the simple line graph depicted in the upper part of Fig. 1 for both of G-ORS and SW-G-ORS. As proposed in [2], the sliding window size is set equal to 10s for SW-G-ORS and SampleRate.

7.1.1 Synthetic Traces

Stationary Environments. We consider three typical scenarios with stationary radio environments as defined in [2]: *steep*, *gradual*, and *lossy*. In steep scenarios, the success probability is either very high or very low. In gradual scenarios, the success probability decreases slowly as the rate increases, and the optimal rate has a success probability higher than 0.5. In lossy scenarios, the best rate has a low success probability, i.e., less than 0.5. We generate three synthetic traces using the success probabilities:

$$\theta_{\text{steep}} = (.99, .98, .96, .93, \mathbf{.90}, .10, .06, .04),$$
$$\theta_{\text{gradual}} = (.95, .90, .80, \mathbf{.65}, .45, .25, .15, .10),$$
$$\theta_{\text{lossy}} = (.90, .80, .70, .55, .45, \mathbf{.35}, .20, .10),$$

for steep, gradual, and lossy scenarios, respectively (the success probability of the optimal rate is highlighted in bold).

Fig. 2 presents the regret of the RA algorithms for each of the three traces. SampleRate explores a new rate at every tenth packet transmission, and hence, SampleRate has a regret increasing lineary with time. G-ORS and SW-G-ORS explore rates in an optimal manner, and significantly outperform SampleRate. The regret of G-ORS grows sub-linearly with time as shown in Theorem 5.2. The regret difference between G-ORS and SW-G-ORS is small and hence the use of a sliding window does not seem to be very detrimental in stationary environments. Observe that according to Theorem 5.1, the optimal rate in the steep scenario can be identified with a lower regret (if one uses an efficient algorithm), i.e., $c_G(\theta_{\text{steep}}) \leq c_G(\theta_{\text{gradual}})$ and $c_G(\theta_{\text{steep}}) \leq c_G(\theta_{\text{lossy}})$. This observation is confirmed by our experiments.

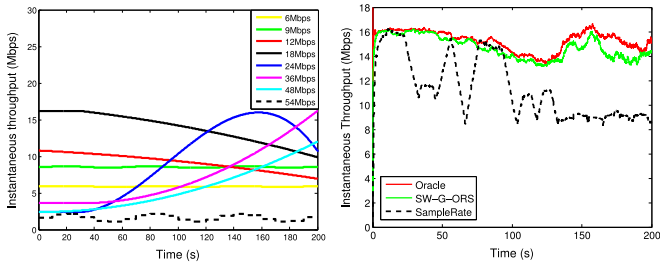
Non-Stationary Environments. We generate a non-stationary trace with time-varying success probabilities $\theta(t)$, which smoothly evolve from "steep" to "gradual" to "lossy" as depicted in Fig. 3a. In Fig. 3b, we plot the performance of SW-G-ORS, Oracle and SampleRate. SW-G-ORS exhibits a performance almost equal to that of the Oracle algorithm and significantly outperforms SampleRate. The performance of SampleRate is particularly poor in the last phase corresponding to a lossy scenario. This is because SampleRate excludes a rate whose four recent successive transmissions have failed even if it is the best rate, and hence it cannot perform well in lossy environments.

7.1.2 Test-Bed Traces

We now present the results obtained from the traces of our 802.11g test-bed, consisting of two 802.11g nodes (SparkLAN WPEA 123AG/E) connected in ad-hoc mode. We collect two traces: (a) to generate a stationary environment, we fix the positions of the two nodes so that the successful packet transmission probabilities are roughly constant over time; (b) to generate a non-stationary environment, we move the receiver at the speed of a pedestrian. The traces record the packet loss history at each rate (we successively send multiple packets of size 1500 bytes, using the 8 available rates in a round robin manner). The traces and results are presented in Fig. 4. Again, in both scenarios, SW-G-ORS clearly outperforms Sample Rate, and exhibits a performance almost equal to that of the Oracle algorithm.

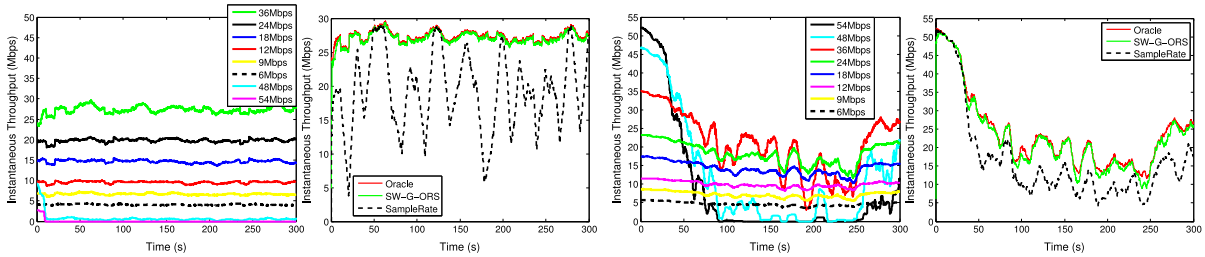
7.2 802.11n MIMO Systems

Next, we investigate the performance of SW-G-ORS in 802.11n systems, supporting both MIMO transmission and packet aggregation. We consider 16 (mode, rate) pairs with two MIMO modes, SS and DS, as in [8], [10], and an aggregated Medium Access Control (MAC) protocol data unit (A-MPDU) frame consisting of 30 subframes of size 1KB each. We plot the performance of four RA algorithms: SW-G-ORS, Sample Rate [2], MiRA [10] and Oracle. For SW-G-ORS, we use the graph G depicted in the lower part of Fig. 1. The size of the sliding window for SW-G-ORS and SampleRate is taken equal to 1 second for fair comparison. It is noted that MiRA is specifically designed for 802.11n systems [10] and jointly performs both rate adaptation and packet aggregation. For a fair comparison, we extract the rate adaptation algorithm from MiRA. It is noted that the design of MiRA is based on some assumptions on the problem structure: a locally *monotonic structure* is assumed (see Section 3.1.2) so that so that transmission at a



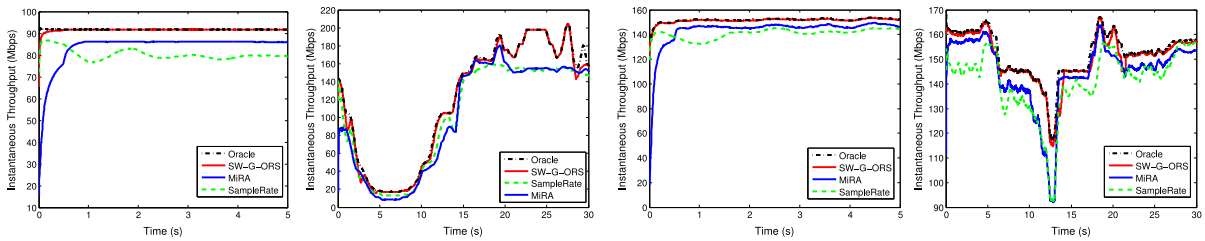
(a) Generated non-stationary trace (b) Average throughput

Fig. 3. Artificially generated non-stationary scenario: (a) The generated non-stationary trace evolving from "steep" to "gradual" to "lossy"; and (b) throughput of SW-G-ORS, SampleRate, and Oracle.



(a) Stationary environment (b) Non-stationary environment

Fig. 4. 802.11g test-bed traces and results: (left) Instantaneous throughput of every rate, and (right) instantaneous throughput of the various RA algorithms in (a) stationary and (b) non-stationary environments.



(a) Artificial traces (b) Test-bed traces

Fig. 5. Instantaneous throughput of the various RA algorithms in 802.11n systems: (left) Stationary environments, and (right) non-stationary environments along (a) artificial traces and (b) test-bed traces.

higher rate (using the same MIMO mode) results in a lower success transmission probability. If this assumption fails, MiRA may fail to identify the optimal decision. MiRA samples the various decisions and computes confidence intervals for the success probability of each decision. To do so, the expectation and standard deviation are estimated using EWMAs with discount factors $\alpha = 1/8$ and $\beta = 1/4$, respectively. Once the best rate for each mode has been identified with high confidence, MiRA zig-zags between neighboring modes to find the best mode. We implement the RA algorithm of MiRA using the same choice of parameters as suggested in [10].

We perform trace-based benchmarks, using both synthetic and test-bed traces. To generate synthetic traces, we use a mapping between the channel measurement (SNR and diffSNR)[2] and the success probability. This mapping was obtained by the authors of [8]. Namely, given decision d, its success probability is given by a known function of d, SNR and diffSNR (the latter two do not depend on d). To generate a stationary environment, we fix a value of SNR and diffSNR, and generate packet losses using the corresponding success probabilities. To generate a non-stationary environment, we vary the values of SNR and diffSNR smoothly, and generate packet losses using the corresponding evolving success probabilities. We also directly use test-bed traces which were used to obtain the mapping mentionned above in [8]. Here each trace corresponds to a different stationary environment. We generate a non-stationary trace by smoothly concatenating 5 stationary traces.

The results are presented in Fig. 5, where the throughput at a given time is computed by averaging over a time window of 0.5 seconds. In stationary environments, while every RA algorithm is able to find the best decision given enough time, SW-G-ORS learns the best decision much faster than MiRA and SampleRate, since these algorithms do not explore decisions in an optimal way. In non-stationary scenarios, the throughput of SW-G-ORS is very close to that of the Oracle. MiRA and SampleRate are more or less able to track the best decision but the performance loss with respect to SW-G-ORS is significant.

2. diffSNR is the maximal gap between the SNRs measured at the various antennas.

8 Test-Bed Experiments

To assess the practical gains obtained by G-ORS algorithms, we conduct experiments in an indoor 802.11n test-bed. The performance of G-ORS algorithms is compared to that of (i) Minstrel HT [41], the default RA algorithm implemented in the 802.11n wireless driver of the current linux kernel, (ii) Atheros MIMO RA, often included in the linux kernel as an alternative option, and (iii) MiRA [10], which has recently been made publicly available [42]. MiRA has been briefly described in the previous section (Section 7.2). The RA algorithms in Minstrel HT and Atheros MIMO are similar: they estimate the expected throughputs achieved at the various (mode, rate) pairs using EWMAs, and probe randomly selected (mode, rate) pairs to track the optimal pair. In Minstrel HT, new pairs are probed periodically and when packet losses are observed at the current best empirical (mode, rate) pair. In Atheros MIMO RA, new pairs are also probed periodically, when the throughput at the current best empirical (mode, rate) pair goes below a given threshold. This contrasts with G-ORS which optimally probes adjacent (mode, rate) pairs, and exploits the unimodal structure efficiently, as highlighted by our theoretical analysis.

8.1 Implementing G-ORS Algorithms

We implement G-ORS algorithms by modifying Minstrel HT [41], a popular open source RA algorithm for 802.11n systems. Using Minstrel as a foundation, we simply modify its RA module and we leave all the other Minstrel modules unchanged, including the frame aggregation module and the adaptive RTS/CTS module.

Selecting the Graph G. To implement G-ORS algorithms, we first need to choose a graph G such that the throughput is a graphically unimodal function w.r.t. G. According to the regret analysis presented earlier, we can achieve a better

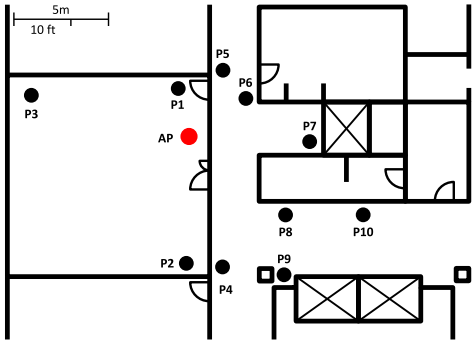


Fig. 6. Floorplan of the indoor 802.11n test-bed used for scenarios (a) and (b).

performance if we select the sparsest of such graphs. Note however, the graphical unimodality of the throughput function is critical for the algorithm to work well; hence it is safer to select a denser graph G – this leads to a more robust algorithm. In our implementation, we add some redundant edges to the graph described in Fig. 1. More precisely, in the implemented graph G, a (mode, rate) pair is connected to at most 8 closest pairs: 4 with higher rates, and 4 lower rates with a tie-breaking rule of favoring DS over SS. For example, the neighbors of $(\mathrm{SS}, 13.5)$ are $\{(\mathrm{SS}, 27), (\mathrm{DS}, 27), (\mathrm{SS}, 40.5), (\mathrm{SS}, 54)\}$, and the neighbors of $(\mathrm{SS}, 108)$ are $\{(\mathrm{DS}, 81), (\mathrm{SS}, 81), (\mathrm{DS}, 54), (\mathrm{SS}, 54)\} \cup \{(\mathrm{DS}, 108), (\mathrm{SS}, 121.5), (\mathrm{SS}, 135), (\mathrm{DS}, 162)\}$.

Efficient Implementation of a Sliding Window. Implementing a real sliding window as that described in SW-G-ORS requires to store and process every transmission in the corresponding time window; and the number of transmissions per second can be up to several thousands. In practice, it is wiser to discount previous transmissions based on the time they occurred. Indeed, this allows to maintain simple indexes for each decision, as done by EWMA-G-ORS, and removes the need to store in memory the outcomes of all transmissions that occurred within a time window. For example, EWMA-G-ORS and MiRA [10] are based on discounting past transmissions at an exponential rate. But roughly speaking, the way that transmissions are discounted does not significantly impact the performance in practice provided that the "average" discount factor is preserved [36]. Here, we have an additional difficulty: transmissions do not occur regularly over time, since the duration of a transmission depends on the chosen rate, and it is important that a transmission is discounted w.r.t. to *time*, rather than its sequence number (as done in EWMA-G-ORS). This is because sliding windows are implemented to cope with non-stationary channel conditions that vary over *time*.

To circumvent this issue, we propose the following approximate implementation of SW-G-ORS. The main idea underlying this approximate implementation is to assume packet transmissions using a decision d are regularly spaced over time. Let t_n denote the time at which we receive the feedback for the nth transmitted frame (A-MPDU packet), and let $\eta_d(n)$ denote the number of packets or sub-frames in the nth frame sent using decision d. It is noted that packets within the same frame use the same decision d, so that $\eta_d(n)$ is the number of packets of the nth frame if the latter uses d, and is equal to 0 otherwise. Finally, as previously let $t_d^\tau(n)$ be our approximate number of packet transmissions using d between time $t_n - \tau$ and t_n. To express $t_d^\tau(n)$ as a function $t_d^\tau(n-1)$, assume that the packet transmissions $t_d^\tau(n-1)$ using d are regularly spaced over time within time interval $(t_{n-1} - \tau, t_{n-1})$. With this assumption, we get:

$$t_d^\tau(n) = \left(1 - \frac{t_n - t_{n-1}}{\tau}\right) t_d^\tau(n-1) + \eta_d(n).$$

The other relevant statistics $\hat{\mu}_d^\tau(n)$ and $l_d^\tau(n)$ are updated in the same manner. For simplicity, until the end of this section, we refer to this new approximate algorithm as "SW-G-ORS", whenever doing so does not create confusion.

8.2 Test-Bed Setup

The indoor 802.11n test-bed consists of one Access Point (AP) and ten clients in the ICT building of KAIST in South Korea as depicted in Fig. 6. The AP and clients are implemented on a desktop computer and laptops, respectively. All of them are operated by Ubuntu 12.04 LTS with Linux kernel 3.11.6. They are equipped with the Qualcomm Atheros AR9280 2.4/5 GHz chipset supporting 802.11n 2×2 MIMO, where both SS and DS MIMO modes are available. Note that since several basic functionalities such as floating point arithmetic and computation of logarithms are unavailable in kernel mode, we have implemented data structures and functions from scratch to allow us to use such basic functionalities.

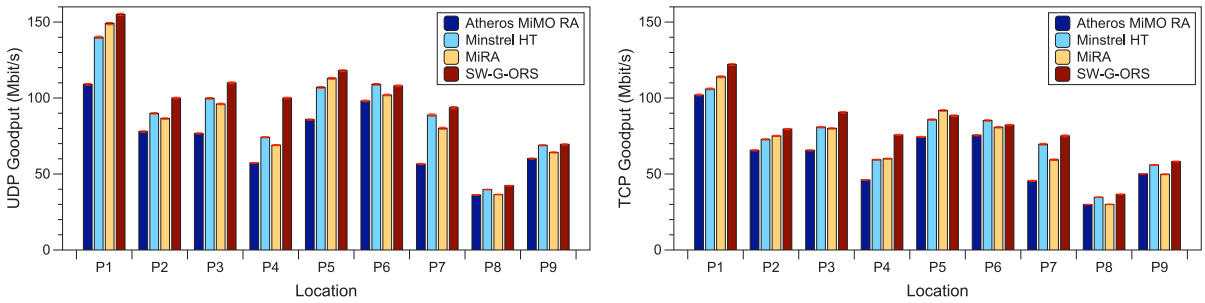
To avoid external interference, we use the 5.4 GHz frequency band, which is typically not available in South Korea.[3] We generate both UDP and TCP network traffic. We use the popular network measurement tool iperf [43] with default settings, where TCP sessions have a packet length of 128 KB, and UDP sessions have a packet length of 8 KB and a sufficient injection rate of 200 Mbit/s. More details about the traffic scenarios are provided below.

8.3 Experiment Results

We evaluate the various RA algorithms in three scenarios, depending on the number of interfering links, and on the nature of the evolving channel conditions (stationary versus non-stationary). We report averaged performance (UDP/TCP goodputs) from 200 measurements over 100s with 95 percent confidence intervals.

(a) Stationary Single Link. In this scenario, the AP sends data packets to 9 clients. The positions of all nodes are fixed as shown in Fig. 6. We measure the average UDP and TCP goodput over 100s on each of the 9 downlink streams (from AP to clients). The results are plotted in Fig. 7. For almost all links, SW-G-ORS outperforms the other RA algorithms for both UDP and TCP traffic. The superiority of SW-G-ORS is flagrant for the link AP-P4. For this link, the other algorithms cannot find the optimal (mode, rate) pair possibly due to the fact that they assume a monotonic structure, which sometimes does not hold in practice. More specifically, MiRA assumes local monotonicity, so that transmission at a higher rate (using the same MIMO mode) results in a lower success transmission probability, and other algorithms assume global monotonicity, so that transmission at a higher rate (using any MIMO mode) results in a lower success transmission probability. On the contrary, SW-G-ORS manages to identify the best pair in all scenarios, and it is the only algorithm to find the best pair for AP-P4. For this link, SW-G-ORS provides a 34 percent

3. We were granted the permission to perform this experiment by the Korean Communications Commission.



(a) UDP traffic (b) TCP traffic

Fig. 7. Test-bed experiments, stationary single link scenario: Goodput of (a) UDP and (b) TCP sessions on each link with 95 percent confidence intervals.

UDP goodput gain over Minstrel HT, 44 percent over MiRA and 74 percent over Atheros MIMO RA.

(b) Non-Stationary Single Link. In this scenario, we move a client from position P5 to position P4 back and forth, while the AP remains static. The client moves at the speed of 1m/s[4] and the travel between P4 and P5 (one-way) takes 20 seconds. Fig. 8 shows the instantaneous UDP goodput under each RA algorithm. All RA schemes except Atheros MIMO seem to quickly adjust the data rate to track the channel variations. Under Atheros MIMO RA, new (mode, rate) pairs are probed when the throughput using the current pair decreases. Hence the (mode, rate) pair does not change when the channel conditions improve, and one has to wait for the next round of probing. We see that SW-G-ORS tracks the optimal (mode, rate) pair in a highly efficient manner. SW-G-ORS achieves the highest average goodput over the experiment with gains up to $13 percent$ over Minstrel HT, up to 25 percent over MiRA and up to 43 percent over Atheros MIMO RA (refer to Table 1 for the average throughput).

(c) Stationary Multiple Links. In this scenario, we consider three interfering transmitters, as shown in Fig. 9a. The network topology corresponds to the "flow-in-the-middle" situation: link (S2-R2) in the middle interferes with both (S1-R1) and (S3-R3) links, whereas the two latter links do not interfere with each other. Fig. 9 shows the average (over 100s) goodputs of the three links under the various RA schemes for both UDP and TCP traffic. It is noted that the y-axis of Fig. 9 has two different labels: one for the goodput of individual links on the left, and another one for the total goodput on the right. SW-G-ORS provides not only the best total performance, but also the best fairness. In scenarios with interference, it is often important to distinguish the cause of transmission failures: collision or channel noise, especially when the used RA algorithm assumes a monotonic structure. Indeed, this assumption may result in very poor performance since it gives rise to a malicious feedback cycle: collisions are misinterpreted as channel errors, so that the RA algorithm decreases the rate, and even more collisions occur. To address this problem, various methods explicitly distinguishing collisions and channel errors have been employed in CARA [17], RRAA [18], H-RCA [15], and MiRA [10], while they keep the monotonic structure assumption (that leads to mistakes when identifying the optimal rate as shown in Fig. 7, AP-P4). SW-G-ORS does not assume a monotonic structure, but rather exploits the robust unimodal structure described through the graph G with redundant connectivity. It avoids the malicious feedback cycle and always finds the optimal rate. At the same time, SW-G-ORS uses the adaptive RTS mechanism of Minstrel HT which enables RTS/CTS handshake for a while once it experiences a burst subframe loss in an A-MPDU packet since the burst subframe loss can be interpreted as a collision [10]. This adaptive RTS mechanism provides a good trade-off between reducing collisions and RTS/CTS overhead. In turn, as illustrated in Figs. 9b and 9c, SW-G-ORS yields very significant UDP/TCP goodput gains particularly for the middle link (S2-R2).

4. In order to reproduce the same mobility pattern for every trial and hence to be able to compare the various RA algorithms, this speed is that of a slow pedestrian (we were dragging a cart containing the client).

9 Conclusion and Perspectives

In this paper, we formulated the design of RA algorithms in 802.11 systems as an online stochastic optimization problem, and more precisely as a structured Multi-Armed Bandit

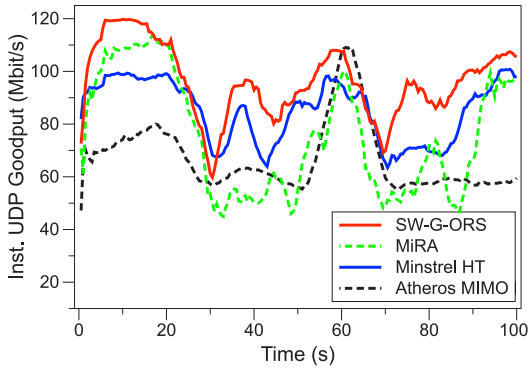
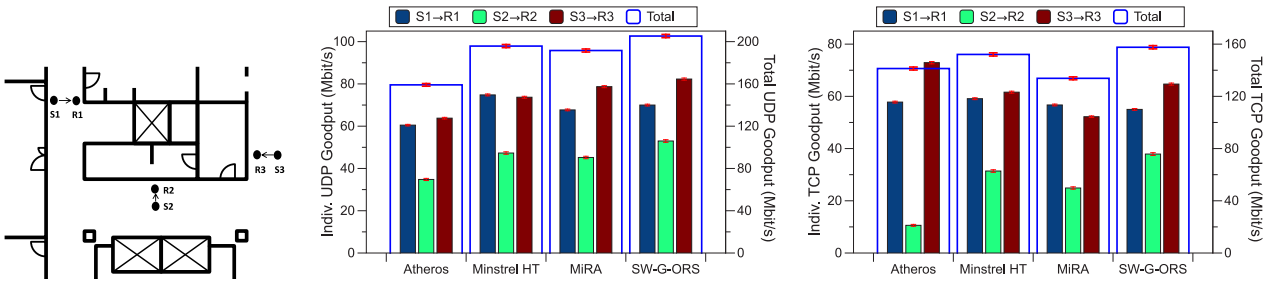


Fig. 8. Test-bed experiments, non-stationary single link scenario: Instantaneous goodput of UDP sessions of each RA algorithm.

TABLE 1
Test-Bed Experiments, Non-Stationary Single Link Scenario: Throughput over 200 Measurements Taken over 100 Seconds and the Corresponding Gains with 95 Percent Confidence Intervals

Rate Adaptation Algorithm	Average UDP Goodput (Mbit/s)	Average Gain (%)
Atheros MIMO RA	67.1 ± 1.80	46.4 ± 3.19
Minstrel HT	85.2 ± 1.60	13.6 ± 1.49
MiRA	76.4 ± 3.05	34.7 ± 3.97
G-ORS	96.2 ± 1.88	-



(a) Floorplan for the non-stationary multiple link scenario

(b) UDP Traffic

(c) TCP Traffic

Fig. 9. Test-bed experiments, stationary multiple links scenario: Goodput of (a) UDP and (b) TCP sessions on each link with 95 percent confidence intervals.

(MAB) problem. This versatile formalism allowed us to devise the family of G-ORS algorithms which includes SW-G-ORS and EWMA-G-ORS, a family of algorithms which provably learn the best decision for transmission as fast as it is possible. Through extensive numerical and test-bed experiments, we have shown that our algorithms outperform state-of-the-art sampling-based RA algorithms in 802.11g and 802.11n systems, including MiRA and Minstrel HT. The design of the latter algorithms were based on heuristics, and were engineering solutions tailor-made for 802.11n systems, in particular, to handle MIMO mode and rate selection and packet aggregation schemes jointly, while G-ORS is generic based on rigorous mathematical arguments.

In G-ORS and SW-G-ORS, we have (rigorously) optimized the MIMO mode and rate selection scheme only. The MAB framework presented here can be used to extend G-ORS to an optimal joint packet aggregation and RA algorithm, which we believe would provide an even higher performance gain compared to MiRA and Minstrel HT. Optimizing packet aggregation and RA schemes jointly would correspond to a unimodal MAB problem with *switching costs*.[5] The switching cost would indeed model the loss in performance due to the overhead (e.g., due to acknowledgments) incurred when changing frame. Other extensions are worth considering. G-ORS and SW-G-ORS can be directly applied to 802.11ac systems if the Multi-User(MU) MIMO feature is not used. However, when one wishes to exploit the MU MIMO feature, we believe that again a MAB framework can be used to devise optimal MU MIMO rate adaptation algorithms.

APPENDIX

Proof of Lemma 3.1. Let $T > 0$. By time T, we know that there have been at least $\lfloor Tr_1 \rfloor$ transmissions, but no more than $\lceil Tr_K \rceil$. Also observe that both regrets R^π and R_1^π are increasing functions of time. We deduce that:

$$R^\pi(\lfloor Tr_1 \rfloor) \leq R_1^\pi(T) \leq R^\pi(\lceil Tr_K \rceil).$$

Now

$$\begin{aligned}\liminf_{T\to\infty} \frac{R_1^\pi(T)}{\log(T)} &\geq \liminf_{T\to\infty} \frac{R^\pi(\lfloor Tr_1 \rfloor)}{\log(T)} \\ &= \liminf_{T\to\infty} \frac{R^\pi(\lfloor Tr_1 \rfloor)}{\log(\lfloor Tr_1 \rfloor)} \geq c.\end{aligned}$$

The second statement can be derived similarly.

5. Switching costs have been addressed in MAB problems without structure, see e.g., in [44], but not in MABs with structure.

Proof of Theorem 5.1. We apply the theory of controlled Markov chains developed in [45]. Using the same terminology and notation as in [45], the parameter θ takes values in $\mathcal{U}_G$; the Markov chain has values in $\mathcal{S} = \{0, r_1, \ldots, r_D\}$; the set of control laws is $\{1, \ldots, D\}$, i.e., each control law corresponds to a (mode, rate) pair; the transition probabilities are given as follows: for all $x, y \in \mathcal{S}$,

$$p(x, y; d, \theta) = p(y; d, \theta) = \begin{cases} \theta_d, & \text{if } y = r_d, \\ 1 - \theta_d, & \text{if } y = 0; \end{cases}$$

finally, the reward $r(x, d) = x$.

We now fix $\theta \in \mathcal{U}_G$. Define $I^d(\theta, \lambda) = I(\theta_d, \lambda_d)$ for any d. Further define the set $B(\theta)$ consisting of all *bad* parameters $\lambda \in \mathcal{U}_G$ such that d^* is not optimal under parameter λ, but which are statistically *indistinguishable* from θ:

$$B(\theta) = \{\lambda \in \mathcal{U}_G : \lambda_{d^*} = \theta_{d^*}, \max_d r_d \lambda_d > r_{d^*} \lambda_{d^*}\},$$

$B(\theta)$ can be written as the union of sets $B_d(\theta)$, $d = 1, \ldots, D$ defined as: $B_d(\theta) = \{\lambda \in B(\theta) : r_d \lambda_d > r_{d^*} \lambda_{d^*}\}$. Note that $B_d(\theta) = \emptyset$ if $r_d < r_{d^*} \theta_{d^*}$. Define $P = \{d : r_d \geq r_{d^*} \theta_{d^*}\}$, and $P' = P \setminus \{d^*\}$.

Let $\pi \in \Pi$ be a uniformly good algorithm. By applying Theorem 1 in [45], we know that $\limsup_T R^\pi(T)/\log(T) \geq c(\theta)$, where $c(\theta)$ is the minimal value of the following optimization problem:

$$\min \sum_d c_d(\mu_{d^*} - \mu_d) \tag{4}$$

$$\text{s.t.} \inf_{\lambda \in B_d(\theta)} \sum_{l \neq d^*} c_l I^l(\theta, \lambda) \geq 1, \quad \forall d \in P' \tag{5}$$

$$c_d \geq 0, \quad \forall d. \tag{6}$$

Now assume that (5) holds. For any $d \in N(d^*) \cap P'$, for any $\epsilon > 0$, select λ such that $r_d \lambda_d = \mu_{d^*} + \epsilon$, and for any $l \neq d$, $\lambda_l = \theta_l$. Then $\lambda \in B_d(\theta)$, and hence:

$$\sum_{l \neq d^*} c_l I^l(\theta, \lambda) = c_d I\left(\theta_d, \frac{\mu_{d^*} + \epsilon}{r_d}\right) \geq 1.$$

We deduce that $c(\theta) \geq c_G^\epsilon(\theta)$ where $c_G^\epsilon(\theta)$ is the minimal value of the optimization problem:

$$\begin{aligned} &\min \sum_d c_d(\mu_{d^*} - \mu_d) \\ &\text{s.t. } c_d I\left(\theta_d, \frac{\mu_{d^*} + \epsilon}{r_d}\right) \geq 1, \quad \forall d \in N(d^*) \\ &\quad c_d \geq 0, \quad \forall d. \end{aligned}$$

Hence for any $\epsilon > 0$, $c(\theta) \geq \sum_{d \in N(d^\star)} \frac{\mu_{d^\star} - \mu_d}{I(\theta_d, \frac{\mu_{d^\star}+\epsilon}{r_d})}$. We conclude that $\limsup_T R^\pi(T)/\log(T) \geq c_G(\theta)$.

Proof of Theorem 5.2. We provide a sketch of proof only due to space limitations, refer to [40] for a complete proof. Let $T > 0$. The regret of $\pi = G - ORS$ up to time T is:

$$R^\pi(T) = \sum_{d \neq d^\star} (\mu_{d^\star} - \mu_d) \mathbb{E}\left[\sum_{n=1}^{T} \mathbf{1}\{d(n) = d\}\right].$$

We decompose the set $\{d(n) = d\}$ into $A_d(n) = \{d(n) = d, L(n) \neq d^\star\}$ (the leader is not $d^\star$) and $B_d(n) = \{d(n) = d, L(n) = d^\star\}$ (the leader is $d^\star$), and analyze the two corresponding contributions to regret. We have:

$$\sum_{d \neq d^\star} (\mu_{d^\star} - \mu_d) \mathbb{E}\left[\sum_{n=1}^{T} \mathbf{1}\{A_d(n)\}\right] \leq r_{d^\star} \sum_{d \neq d^\star} \mathbb{E}[l_d(T)].$$

Now when $L(n) = d^\star$, G-ORS selects a decision $d \in \mathcal{N}(d^\star)$, we deduce that $R^\pi(T)$ is upper bounded by:

$$r_{d^\star} \sum_{d \neq d^\star} \mathbb{E}[l_d(T)] + \sum_{d \in N(d^\star)} (\mu_{d^\star} - \mu_d) \mathbb{E}\left[\sum_{n=1}^{T} \mathbf{1}\{B_d(n)\}\right].$$

The main difficulty consists in bounding the first term, i.e., the average number of times where $d^\star$ is not the leader. The following result provides the required bound. Its proof presented in [40][Theorem C.1] relies on concentration inequalities, and properties of the KL divergence.

$$\mathbb{E}[l_d(T)] = O(\log(\log(T))), \quad \forall d \neq d^\star.$$

From the above theorem, we conclude that the leader is $d^\star$ except for a negligible number of instants (in expectation). When $d^\star$ is the leader, G-ORS behaves as KL-UCB [39] restricted to the set $N(d^\star)$ of possible decisions. Following the same analysis as in [39], we can show that for all $\epsilon > 0$ there are constants C_1 , $C_2(\epsilon)$ and $\beta(\epsilon) > 0$ such that:

$$\mathbb{E}\left[\sum_{n=1}^{T} \mathbf{1}\{B_d(n)\}\right] \leq \mathbb{E}\left[\sum_{n=1}^{T} \mathbf{1}\{b_d(n) \geq b_{d^\star}(n)\}\right]$$
$$\leq (1+\epsilon) \frac{\log(T)}{I(\theta_d, \frac{r_{d^\star}\theta_{d^\star}}{r_d})} + C_1 \log(\log(T)) + \frac{C_2(\epsilon)}{T^{\beta(\epsilon)}}.$$

Putting pieces together, we get:

$$R^\pi(T) \leq (1+\epsilon) c_G(\theta) \log(T) + O(\log(\log(T))),$$

which concludes the proof.

ACKNOWLEDGMENTS

The authors would like to thank the authors of [8] for sharing their traces with them. Alexandre Proutiere and Richard Combes (during his post-doc at KTH, Stockholm) were supported by the ERC consolidator grant 308267. Alexandre Proutiere's research is further funded by Vetenskapradet (VR), and the Stiftelsen for Strategisk Forskning (SSF). This work was partially supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation. The work of Yung Yi is supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (No. 2016R1A2A2A05921755) and Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2016-0-00160,Versatile Network System Architecture for Multi-dimensional Diversity). A preliminary version of this work has appeared in IEEE INFOCOM 2014.

REFERENCES

[1] A. Kamerman and L. Monteban, "WaveLAN-ii: A high-performance wireless LAN for the unlicensed band," *Bell Labs Tech. J.*, vol. 2, no. 3, pp. 118–133, 1997.

[2] J. Bicket, "Bit-rate selection in wireless networks," Ph.D. dissertation, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 2005.

[3] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level measurements from an 802.11b mesh network," in *Proc. ACM Conf. Appl. Technol. Architectures Protocols Comput. Commun.*, 2004, pp. 121–132.

[4] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, "Measurement-based models of delivery and interference in static wireless networks," *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 4, pp. 51–62, Aug. 2006.

[5] J. Camp and E. Knightly, "Modulation rate adaptation in urban and vehicular environments: cross-layer implementation and experimental evaluation," in *Proc. Annu. Int. Conf. Mobile Comput. Netw.*, 2008, pp. 315–326.

[6] J. Zhang, K. Tan, J. Zhao, H. Wu, and Y. Zhang, "A practical SNR-guided rate adaptation," in *Proc. IEEE 27th Conf. Comput. Commun.*, 2008, pp. 146–150.

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

[8] L. Deek, E. Garcia-Villegas , E. Belding, S.-J. Lee, and K. Almeroth, "Joint rate and channel width adaptation in 802.11 MIMO wireless networks," in *Proc. IEEE Int. Conf. Sensing Commun. Netw.*, 2013, pp. 167–175.

[9] R. Crepaldi, J. Lee, R. Etkin, S.-J. Lee, and R. Kravets, "CSI-SF: Estimating wireless channel state using CSI sampling and fusion," in *Proc. IEEE INFOCOM*, 2012, pp. 154–162.

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

[11] T. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances Appl. Math.*, vol. 6, no. 1, pp. 4–2, 1985.

[12] G-ORS drivers, (2015). [Online]. Available: http://lanada.kaist.ac.kr/ors.html

[13] M. Lacage, M. Manshaei, and T. Turletti, "IEEE 802.11 rate adaptation: A practical approach," in *Proc. 7th ACM Int. Symp. Model. Anal. Simul. Wireless Mobile Syst.*, 2004, pp. 126–134.

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

[15] K. D. Huang, K. R. Duffy, and D. Malone, "H-RCA: 802.11 collision-aware rate control," *IEEE/ACM Trans. Netw.*, vol. 21, no. 4, pp. 1021–1034, Aug. 2013.

[16] Q. Pang, V. Leung, and S. Liew, "A rate adaptation algorithm for IEEE 802.11 WLANs based on MAC-layer loss differentiation," in *Proc. IEEE 2nd Int. Conf. Broadband Netw.*, 2005, pp. 659–667.

[17] J. Kim, S. Kim, S. Choi, and D. Qiao, "CARA: Collision-aware rate adaptation for IEEE 802.11 WLANs," in *Proc. IEEE Int. Conf. Comput. Commun.*, 2006, pp. 1–11.

[18] S. Wong, H. Yang, S. Lu, and V. Bharghavan, "Robust rate adaptation for 802.11 wireless networks," in *Proc. ACM 12th Annu. Int. Conf. Mobile Comput. Netw.*, 2006, pp. 146–157.

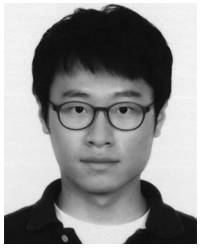
[19] G. Holland, N. Vaidya, and P. Bahl, "A rate-adaptive MAC protocol for multi-hop wireless networks," in *Proc. ACM 7th Annu. Int. Conf. Mobile Comput. Netw.*, 2001, pp. 236–251.

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

[21] G. Judd, X. Wang, and P. Steenkiste, "Efficient channel-aware rate adaptation in dynamic environments," in *Proc. ACM 6th Int. Conf. Mobile Syst. Appl. Services*, 2008, pp. 118–131.

[22] I. Haratcherev, K. Langendoen, R. Lagendijk, and H. Sips, "Hybrid rate control for IEEE 802.11," in *Proc. ACM 2nd Int. Workshop Mobility Manage. Wireless Access Protocols*, 2004, pp. 10–18.
[23] M. Vutukuru, H. Balakrishnan, and K. Jamieson, "Cross-layer wireless bit rate adaptation," in *Proc. ACM SIGCOMM Conf. Data Commun.*, 2009, pp. 3–14.
[24] Third Generation Partnership Project, 3GPP TR 25.848 V 4.0.0, Mar. 2001.
[25] D. Kim, B. Jung, H. Lee, D. Sung, and H. Yoon, "Optimal modulation and coding scheme selection in cellular networks with hybrid-ARQ error control," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 5195–5201, Dec. 2008.
[26] K. Freudenthaler, A. Springer, and J. Wehinger, "Novel SINR-to-CQI mapping maximizing the throughput in hsdpa," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2007, pp. 2231–2235.
[27] B. Radunovic, A. Proutiere, D. Gunawardena, and P. Key, "Dynamic channel, rate selection and scheduling for white spaces," in *Proc. ACM 7th Conf. Emerging Netw. Exp. Technol.*, 2011, Art. no. 2.
[28] S. Bubeck and N. Cesa-Bianchi, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *Foundations Trends Mach. Learn.*, vol. 5, no. 1, pp. 1–122, 2012.
[29] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. pp. 285–294, 1933.
[30] H. Robbins, "Some aspects of the sequential design of experiments," *Bulletin Amer. Math. Soc.*, vol. 58, no. 5, pp. 527–535, 1952.
[31] J. Y. Yu and S. Mannor, "Unimodal bandits," in *Proc. Int. Conf. Mach. Learn.*, 2011, pp. 41–48.
[32] R. Combes and A. Proutiere, "Unimodal bandits: Regret lower bounds and optimal algorithms," in *Proc. 31st Int. Conf. Int. Conf. Mach. Learn.*, 2014, pp. I-521–I-529.
[33] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite time analysis of the multiarmed bandit problem," *Mach. Learn.*, vol. 47, no. 2–3, pp. 235–256, 2002.
[34] L. Kocsis and C. Szepesvári, "Discounted UCB," in *Proc. 2nd PASCAL Challenges Workshop*, 2006.
[35] J. Y. Yu and S. Mannor, "Piecewise-stationary bandit problems with side observations," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, Art. no. 148.
[36] A. Garivier and E. Moulines, "On upper-confidence bound policies for switching bandit problems," in *Proc. 22nd Int. Conf. Algorithmic Learn. Theory*, 2011, pp. 174–188.
[37] A. Slivkins and E. Upfal, "Adapting to a changing environment: The brownian restless bandits," in *Proc. 21st Conf. Learn. Theory*, 2008, pp. 343–354.
[38] A. Slivkins, "Contextual bandits with similarity information," *J. Mach. Learn. Res.*, vol. 19, pp. 679–702, 2011.
[39] A. Garivier and O. Cappé, "The KL-UCB algorithm for bounded stochastic bandits and beyond," in *Proc. Conf. Learn. Theory*, 2011, pp. 359–376.
[40] R. Combes, A. Proutiere, D. Yun, J. Ok, and Y. Yi, "Optimal rate sampling in 802.11 systems," 2013. [Online]. Available: http://arxiv.org/abs/1307.7309
[41] Minstrel HT Linux Wireless, (2013). [Online]. Available:
[42] MiRA website, (2010). [Online]. Available: http://metro.cs.ucla.edu/resources.html
[43] iPerf, (2013). [Online]. Available: https://iperf.fr/
[44] R. Agrawal, M. V, and D. Teneketzis, "Asymptotically efficient adaptive allocation rules for the multi-armed bandit problem with switching cost," *IEEE Trans. Automat. Control*, vol. 33, no. 10, pp. 899–906, Oct. 1988.
[45] T. L. Graves and T. L. Lai, "Asymptotically efficient adaptive choice of control laws in controlled markov chains," *SIAM J. Control Optimization*, vol. 35, no. 3, pp. 715–743, 1997.

Richard Combes received the engineering degree from Telecom Paristech, in 2008, the master's degree in mathematics from the University of Paris VII, in 2009, and the PhD degree in mathematics from the University of Paris VI, in 2013. He is currently an assistant professor with the Centrale-Supelec and L2S. He was a visiting scientist at INRIA (2012) and a post-doc at KTH (2013). He received the best paper award at CNSM 2011. His current research interests include machine learning, networks, and probability.



Jungseul Ok received BS degree from the School of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), in 2011, and the PhD degree under the supervision of Prof. Yung Yi and Prof. Jinwoo Shin, in 2016. He is currently working as postdoctoral researcher with the Department of Automatic Control, KTH Royal Institute of Technology. He is interested in understanding and improving practical systems based on theoretical model and analysis. His research interest include the intersection of applied mathematics and network systems including wireless network, social network, and probabilistic graphical model, where his major focus is social models with certain engineering purposes such as crowdsourcing and information spread.

Alexandre Proutiere received the graduate degree in mathematics from École Normale Supérieure, Paris, France, the engineering degree from Télécom ParisTech, Paris, France, and the PhD degree in applied mathematics from École Polytechnique, Palaiseau, France, in 2003. He is an engineer from Corps of Mines. In 2000, he joined France Telecom R&D as a research engineer. From 2007 to 2011, he was a researcher at Microsoft Research, Cambridge, United Kingdom. He is currently a professor with the Department of Automatic Control at the KTH The Royal Institute of Technology, Stockholm, Sweden. He was the recipient in 2009 of the ACM Sigmetrics Rising Star Award, and received the Best Paper Awards at the ACM Sigmetrics Conference in 2004 and 2010, respectively and at the ACM Mobihoc Conference in 2009.

Donggyu Yun received the BS, MS, and PhD degrees in the electrical engineering from KAIST, South Korea, in 2009, 2011, and 2016, respectively. He is currently with Naver Search Division, Naver, South Korea. His research interests include machine learning, data analysis, and wireless networks.

Yung Yi received the BS and MS degrees from the School of Computer Science and Engineering, Seoul National University, South Korea, in 1997 and 1999, respectively, and the PhD degree from the Department of Electrical and Computer Engineering, University of Texas at Austin, in 2006. From 2006 to 2008, he was a post-doctoral research associate with the Department of Electrical Engineering, Princeton University. Now, he is an associate professor with the Department of Electrical Engineering, KAIST, South Korea. His current research interests include the design and analysis of computer networking and Wireless communication systems, especially congestion control, scheduling, and interference management, with applications in wireless ad hoc networks, broadband access networks, economic aspects of communication networks, and green networking systems. He received the best paper awards at IEEE SECON 2013 and ACM MOBIHOC 2013, and the IEEE William R. Bennett Award 2016.