Monitoring Multi-hop Multi-channel Wireless Networks: Online Sniffer Channel Assignment

Jing Xu[a], Wei Liu[a], Kai Zeng[b]
[a]School of Electronic Infor. and Comm., Huazhong Univ. of Science and Technology, Wuhan 430074, China
[b]Dept. of Electrical and Computer Engineering, George Mason University, Fairfax, VA 22030, U.S.
E-mail:{xujing, liuwei}@hust.edu.cn, kzeng2@gmu.edu

***Abstract*—Data capture is important for some critical network applications, such as network diagnosis and criminal investigation. In multi-channel wireless networks, the fundamental challenge for data capture is how to assign operation channels to wireless sniffers. The existing approaches make some impractical assumptions, such as the prior knowledge on network traffic and the perfect conditions of data capture. In this paper, we relax these assumptions and investigate the sniffer-channel assignment problem in multi-hop scenarios. Especially, sniffer redundancy deployment is discussed, which enables multiple sniffers to monitor one traffic. This problem is formulated as a combinatorial multi-arm bandit (MAB) problem, and a cooperative distribute learning policy is proposed.We analyze the regret of our policy in theory, and validate its effectiveness through numerical simulations.**

I. Introduction

Data capture is an important approach to evaluate network performance, which has been adopted in various applications, such as traffic monitoring, malicious activities detection, and so on. To gather the detailed PHY/MAC information, a passive monitoring framework has been proposed [1], in which a dedicated set of hardware devices, called wireless sniffers are introduced. The sniffers are deployed to capture the wireless signals in their vicinity. Then the gathered information can be analyzed in a centralized or distributive way.

Such sniffer-based monitoring framework has attracted increasing attention, especially in the multi-channel wireless works, such as WLAN, wireless mesh networks, cognitive radio networks. Due to less number of wireless sniffers of wireless sniffers as well as limited capturing capability, one fundamental technical challenge is how to assign the working channels for a limited number of sniffers, i.e. sniffer-channel assignment problem, so as to maximize the total amount of information gathered.

There have been some research focused on this area [2]–[4]. They assume that the statistics for all the users' activities are known or could be inferred. Recently, Arora et al. [5] study this problem without the prior assumptions on the traffic statistics. They propose to sequentially learn the traffic pattern while making dynamic channel assignment decisions. Specially, there exists a tradeoff between *exploitation* (i.e., to assign sniffers exclusively to a few channels that are estimated with maximum rewards) and *exploration* (i.e., to try out on each channel to find the best one).

However, the existing studies imply in a local monitoring region. Extending to a multi-hop scenario, a special issue emerges, named "sniffer redundancy", which is to assign multiple sniffers to monitor a certain object. For example, a node is in the vicinity of different sniffers or a flow will happen on the links located in different places, which is likely to lead to the repetition of captured packets. The similar sniffer redundancy problem has been studied in [5], which is devoted to eliminate the repetition of active monitored user introduced by the spatial overlap of sniffer monitoring regions.

However, the redundancy of sniffers is not always harmful, especially in wireless monitoring environment. In practice, a variety of factors may cause miss capturing, such as poor wireless channel conditions, and sniffer suspension due to operational failure or sleep mode for energy saving. Unreliable monitoring conditions will affect the quality of data capture and learning results. This issue has been investigated in recent work [6]. Since the observation results from the sniffers are independent, the more sniffers are deployed to monitor one object, the more capture uncertainty could be eliminated. Therefore, with the introduction of uncertainty in passive monitoring, another option of sniffer deployment is also introduced. That is, if one sniffer finds a small reward from monitoring current object, one candidate choice is to work together with the other sniffers on a more promising one.

In this paper, we investigate the sniffer-channel assignment problem for data capturing in a multi-hop multi-channel wireless network scenario, in which the users' traffic statistic information is unknown a priori and the monitoring conditions are unreliable. Our objective is to capture the packets of interesting flows as many as possible. Especially, we focus on sniffer redundancy deployment. The reliability of data capture will be improved by assigning multiple sniffers to monitor one data flow. While, sniffer redundancy will also reduce the chances of exploring the best monitoring flow. An interesting tradeoff arises, *exploitation* (i.e. to monitor one flow more accurately with more sniffers) and *exploration* (i.e. to monitor more flows not so clearly with one sniffer for one flow).

To this end, we formulate the problem as a combinatorial multi-armed bandit (MAB) problem [7] and the idea of online learning is inherited. Moreover, the multi-hop scenario generally implies a large scale network, which requires a decentralized implementation. To address it, we propose an efficient policy to solve the combinatorial MAB problem for sniffer-channel assignment. Our main contributions are summarized as follows:

- Sniffer redundancy deployment is investigated to achieve a new trade-off between *exploitation* and *exploration*.
- A cooperative distributed implementation of learning policy is proposed for the multi-hop scenario.

The reminder of this paper is organized as follows. Section II presents our problem and its formulation, and the details of our policy are introduced in Section III. Performance evaluation of our proposed policy is shown in Section IV, followed by conclusion in Section V.

DOI 10.1109/LCN.2016.97

II. Problem Description

A. System model

1) Network scenario: Given a multi-hop multi-channel wireless network $G(V, E, C)$, there are a set V of nodes and a set C of available wireless channels without any inter-channel interference, where $|C| \geq 2$. E is the set of possible communication links. The nodes are scattered in a large scale plane. The adjacent nodes (i.e. within the transmission range of each other) that work on the same channel construct a link $l \in E$. The channel $c_l \in C$ of a link l is chosen according to one of existing channel assignment algorithms in the literature [8]. Similar to [5], the nodes communicate according to a synchronous slot structure. For limited transmission range, when two distant nodes need to communicate, a route with multiple consecutive links will be established and their packets will be transmitted along the path by the means of relay.

2) Wireless monitoring system: Independent of the communication system, we introduce a third-party monitoring system for network G. The objective is to capture the packets of interested flows as many as possible. Whether a flow is of interest or not depends on the purpose of the monitoring system. We assume that S sniffers are deployed in the monitoring system, each of which can switch among multiple channels.

The sniffer deployment could be conducted in two steps. The first step is to solve the spatial coverage problem, which aims to distribute the sniffers to achieve the maximum coverage. After that, the locations of the sniffers are given. Based on the results, the next issue is to determinate the channel of each sniffer, i.e., sniffer-channel assignment. The main reason to separate the two problems is to support dynamical network topology. Once the network topology changes, the system only needs to adjust the monitoring channel of each sniffer. It avoids sniffer location rearrangement that is usually costly and time-consuming. In this paper, we focus on the latter issue.

We assume that the sniffers have capture capability in the frame level such that the packets can be distinguished. But there are some restrictions for the sniffers. Firstly, each sniffer can only listen on one channel at any instant of time, so as to capture the whole packet. Secondly, each sniffer has a limited monitoring range. The link set within the monitor range of the sniffer s is denoted as $\boldsymbol{D}(s)$. The necessary condition that the packets of links can be observed by sniffer s is given by $\boldsymbol{D}(s) \bigcap \{l \mid c_l = k_s\} \neq \emptyset$ if its monitoring channel is k_s. Since the operating channel of each link in a time slot is unique, the sniffer-channel assignment is equivalent to sniffer-link assignment. In the rest part of this paper, we will use the links as objects instead of the channels.

Furthermore, we consider a practical scenario with considerations of unreliable monitoring conditions, where the packets can be successfully captured with a certain probability. We define the probability as capture probability. Without loss of generality, we assume that capture probabilities of the sniffers on each link are heterogeneous, which can be represented as a matrix $\boldsymbol{P}_c$. Each entry $P_c(s, l) \in [0, 1]$ is the average capture probability when sniffer s monitors link l. We assume that $\boldsymbol{P}_c$ is known a priori via measurement and is relatively stable.

B. Problem formulation

Considering a multi-hop network, there are N target flows transmitted along L active links in the monitoring region of the sniffers. For simplicity, we assume that single path routing is adopted in the communication network and the routing of each flow is relatively stable. Let $\boldsymbol{Path} = \{\boldsymbol{path}_1, \boldsymbol{path}_2, \ldots, \boldsymbol{path}_N\}$ be the path set, in which each entry is a dependent set of links. Similar to [5], we assume that the traffic on a given flow f is drawn from an i.i.d Bernoulli stochastic process x_f^t over time t with mean packet occurrence probability $p_f \in (0, 1)$. Accordingly, the link traffic $(x_l^t, l \in [L], t \in [T])$ over time $T > 0$ is also an i.i.d random sequence. And the packet occurrence probability on each link that belongs to a path is equal to that of a flow on the path. That is, $p_l = p_f, \forall\ l \in \boldsymbol{path}_f$. Note that some paths will pass through the same link in some cases. In this case, packet occurrence probability of a flow will be overestimated by using that of the link. Fortunately, the packets from different flows cannot be transmitted on the same link at the same time. Thus, x_l^t can be decomposed as multiple components based on Independent Component Analysis (ICA). Each component is the packet occurrence probability corresponding to a flow. The link belongs to multiple paths can be deemed as multiple virtual links, each of which bears a different flow.

Let $\mathcal{A}$ be the set of admissible deployment of sniffers, $\mathcal{A} = \{\boldsymbol{a} = (a_{s,l} \in \{0, 1\}), \forall s \in [S], l \in [L]\}$, where $[L] = \{1, 2, \ldots, L\}, [S] = \{1, 2, \ldots, S\}$[1]. For any $\boldsymbol{a} \in \mathcal{A}$, $a_{s,l} = 1$ if sniffer s is assigned to monitor link l, and 0 otherwise. Our objective is to maximize the expected number of packets captured given any traffic pattern of target flows.

Different from the existing works, in our formulation, when a particular deployment scheme $\boldsymbol{a}^t$ is selected, packets will only be captured with a probability. In particular, given any realization of link traffic $(x_l^t, l \in [L], t \in [T])$ over time $T > 0$, let $y_{s,l}^t$ be the observation of sniffer s on link l at time slot t.

$$y_{s,l} = \begin{cases} x_l^t, & \text{w.p. } P_c(s,l) \\ 0, & \text{w.p. } 1 - P_c(s,l) \end{cases}$$

Two special cases require us to calculate the reward function carefully with the introduction of capture probability. One is sniffer redundancy on a link. It implies that multiple sniffers monitor a link at the same time. We can represent the capture probability of link l as

$$\phi_l(\boldsymbol{a}) = 1 - \prod\nolimits_s [(1 - P_c(s,l))^{a_{s,l}}]$$

The other case is sniffer redundancy on a flow. That is, multiple sniffers are deployed to monitor different links that belong to the same path of a flow. Thus, it requires us to calculate the reward based on flows rather than links. From the view point of traffic flow, capture probability of the packets of a flow f depends on sniffer deployment of all the links that belong to it. Thus, the capture probability of flow f is

$$\phi_f(\boldsymbol{a}) = 1 - \prod\nolimits_{l \in \boldsymbol{path}_f} \prod\nolimits_s [(1 - P_c(s,l))^{a_{s,l}}]$$

Obviously, $\phi_f = 0$ when $a_{s,l} = 0, \forall s \in [S], l \in \boldsymbol{path}_f$.

We define the reward $R(\boldsymbol{a}^t)$ of using $\boldsymbol{a}^t$ at time t as

$$R_f(\boldsymbol{a}^t) = \begin{cases} x_f^t, & \text{w.p. } \phi_f(\boldsymbol{a}^t) \\ 0, & \text{w.p. } 1 - \phi_f(\boldsymbol{a}^t) \end{cases}$$

and

$$R(\boldsymbol{a}^t) = \sum\nolimits_f R_f(\boldsymbol{a}^t).$$

[1]Other notation in the form of $[\cdot]$ is similarly defined.

The decision $\boldsymbol{a}^t$ is based on all past observations $\boldsymbol{h}^t = (y_{s,l}^{\varsigma}, s \in [S], l \in [L], \varsigma \in [t-1])$. Let the set of all possible observations over time be $\mathcal{H}$, and the decision policy of the monitoring system is then given by a map $\sigma : \mathcal{H} \rightarrow \mathcal{A}$. Given *any* realization of the traffic $(x_l^t, l \in [L], t \in [T])$, the reward of any policy σ is defined as

$$G(\sigma) := \mathbb{E}^{\sigma}\left\{\sum\nolimits_{t=1}^{T} R(\boldsymbol{a}^t)\right\},$$

where the expectation is taken with respect to the randomness in capture process, which depends on the choice of σ, and we explicitly denote this dependency using the superscript.

We evaluate a given policy σ by using the notion of regret, which is defined as the difference between the average number of packets captured by the dynamic deployment using σ, and that by the optimal static sniffer-channel assignment scheme $\boldsymbol{a}^*$ in hindsight. Formally, given $(x_l^t, l \in [L], t \in [T])$, let

$$\boldsymbol{a}^{\boldsymbol{*}} = \arg\max_{\boldsymbol{a} \in \mathcal{A}} \mathbb{E}^{\boldsymbol{a}}\left\{\sum\nolimits_{t=1}^{T} R(\boldsymbol{a})\right\}.$$

Similarly, the expectation is taken with respect to the randomness of packet capture given $\boldsymbol{a}$. Let $\sigma^* : \mathcal{H} \rightarrow \mathcal{A}$ be a static policy such that the deployment $\boldsymbol{a}^*$ is eternal for any $\boldsymbol{h}^t \in \mathcal{H}$, and set $G_{\max} = G(\sigma^*)$. The regret is then defined as $Regret = G_{\max} - G(\sigma)$.

III. Online Sniffer-channel assignment

A. A distributed implementation

In general, the multi-hop network implies a large scale scenario. The centralized architecture requires a coordinator who globally collects the observed packet information, centrally computes the decision scheme, and informs the sniffers to assign their monitoring objectives. Such approach is not preferable in the large scale scenario. Thus, we inherit classic LLR policy [9] designed for the combinatorial MAB problem and give its distributed implementation for the sniffer-channel assignment in the multi-hop scenario.

Different from the conventional distributed learning policies, we assume that the sniffers are cooperative. There is a network that enables the sniffers to communicate with each other for sharing all of their learning results periodically. The communication period lasts for τ time slots. In each round, each sniffer announces a special message to all the other sniffers, named StaMsg. In order to reduce communication overhead, the StaMsg contains $(\eta, \langle M_f^s(\eta), \tilde{\lambda}_f^s(\eta)\rangle)$ instead of original packets, in which η is designed to assure information synchronization of the process and $\langle M_f^s, \tilde{\lambda}_f^s\rangle$ is the corresponding statistical information vector of all monitored flows in the vicinity of sniffer s. Through message exchange, the sniffers can get "global" information from each other. Note that the "global" information will be obtained after τ time slots. To omit the impact of this delay, we wrap up every τ consecutive time slots as one round. See Algorithm 1 for the main framework of our distributed learning policy.

The ηth round begins at time slot $(\eta-1)\tau$ and ends at time slot $\eta\tau$. At the beginning phase of the η-th batch cycle, each sniffer gets $\tilde{\lambda}_f$ by merge operation once receiving messages from the other ones (step 4). The messages sent over the monitoring network will incur a non-negligible latency. The latency is proportional to the distance between the sniffers, and the longest possible latency is therefore proportional to the diameter of the monitoring network, denoted as $\tau_0 (\leq \tau)$. In the merge process (step 8), the statistic information $\tilde{\lambda}_f$ with more M_f is selected as the final result of a flow. Then a sniffer-channel assignment $\boldsymbol{a}^{\eta}$ is decided by invoking the scheme decision (step 13-14). After that, no message exchange will happen and the sniffer-channel assignment schemes on each time slot in a round are consistent (step 15). That is, the sniffers are deployed according to the scheme for the remained time slots, i.e. $\boldsymbol{a}^{(\eta-1)\tau+\tau_0+1} = \boldsymbol{a}^{(\eta-1)\tau+\tau_0+2} = \ldots = \boldsymbol{a}^{\eta\tau}$. At the end of the round, the packet capture reward on each time slot is accumulated and the average value is calculated for the update of the statistical means (step 18-19).

Algorithm 1 distributed online sniffer-channel assignment

Input:
Flow number: N; Link number: L; Sniffer number: S;
The dependent link set along the route of flow f: $\boldsymbol{Path}_f$;
Communication period: τ; the longest exchange delay: τ_0

1: **for** for each round $\eta = 1, 2, ..., T/\tau$ **do**
2: **for** $s \in [S]$ **do**
3: **for** each time slot $t = 1, 2, \ldots, \tau$ **do**
4: waits message from other sniffers till τ_0 ends
5: **if** receives a message from sniffer s' **then**
6: verifies the message is newer than the local record by comparing $\eta_{s'}$
7: **if** the message is newer **then**
8: updates statistical means of packet occurrence probabilities of the corresponding flows

$$\tilde{\boldsymbol{\lambda}}_f^s(\eta-1) = \text{Merge}(\tilde{\lambda}_f^s(\eta-1), \tilde{\lambda}_f^{s'}(\eta-1))$$

9: **else**
10: discards the received messages
11: **end if**
12: **end if**
13: calculates estimated packet occurrence probability

$$\tilde{p}_f^s(\eta) = \tilde{\lambda}_f^s(\eta) + \sqrt{\frac{(\sum\limits_f \mathbb{I}(\sum\limits_{l \in \boldsymbol{path}_f} \sum\limits_s a_{s,l}^t > 0) + 1)\ln t}{M_f^s(\eta-1)}}$$

14: strategy decision: $\boldsymbol{a}_s^t = \text{Decision}(\tilde{\boldsymbol{p}}^{\boldsymbol{s}}(\eta))$
15: maintains the sniffer channel assignment scheme at the remained time slots of current period

$$\boldsymbol{a}_s^t = \boldsymbol{a}_s^{t-1}$$

16: waits packet captured results $\boldsymbol{Y_l}(t)$
17: **end for**
18: calculates the average reward

$$y_f(\eta) = \frac{1}{\tau - \tau_0} \sum_t y_l(t), l \in \boldsymbol{path_f}$$

19: updates $\boldsymbol{M}^s(\eta)$ and $\tilde{\boldsymbol{\lambda}}^s(\eta)$:

$$M_f^s(\eta) = \begin{cases} M_f^s(\eta-1)+1, & \text{if } \sum\limits_{l \in \boldsymbol{path}_f} \sum\limits_s a_{s,l}^t > 0 \\ M_f^s(\eta-1), & \text{else} \end{cases} \quad (1)$$

$$\tilde{\lambda}_f^s(\eta) = \begin{cases} \frac{\tilde{\lambda}_f^s(\eta-1) \times M_f^s(\eta-1) + \frac{Y_f(\eta)}{\phi_f(\boldsymbol{a}^t)}}{M_f^s(\eta)}, & \text{if } \sum\limits_{l \in \boldsymbol{path}_f} \sum\limits_s a_{s,l}^t > 0 \\ \tilde{\lambda}_f^s(\eta-1), & \text{else} \end{cases} \quad (2)$$

20: broadcasts StaMsg message
21: **end for**
22: **end for**

In the decision process (step 14), each sniffer utilizes the estimated packet occurrence probabilities of all flows to select a sniffer-channel assignment scheme with the maximum expected reward. Obviously, the problem is a non-linear integral programming, which has been proven to be NP-hard. Fortunately, the reward function is monotony and the key property can be exploited to solve it with a polynomial-time algorithm. The details of the algorithm can be referred to [10].

Note that all the links are deemed as arms in our formulation. We further divide the links into different clusters. The links belong to the same path are in one cluster. The update of statistical information is based on clusters/flows. Two vectors with the size of N are maintained to store and update the statistical means of all flows' packet transmission.

B. Regret analysis

Our policy is built on the LLR policy. Specially, the LLR policy can be regarded as the non-delayed version, where τ is set to be 0. To this end, we conclude the regret guarantees of non-delayed version by referring to that of LLR policy.

LEMMA 1. *The expected regret of the non-delayed version of our policy over any sequence trials is bounded by*

$$Regret_0(T) \leq [\frac{4(\delta_{max})^2 S^2(S+1)N\ln T}{\Delta^2_{min}} + N + \frac{\pi^2}{3}SN]\Delta_{max},$$

where Δ_{min} and Δ_{max} is minimum and maximum of gap between $R(\boldsymbol{a}^*)$ and the reward of non-optimal scheme, $\delta_{max} = \underset{f}{max}\phi_f(\boldsymbol{a}), \forall f \in [N]$.

Based on the lemma, we can easily deduce the regret guarantees of Algorithm 1 as following.

THEOREM 1. *The expected regret of our policy over any sequence trials is bounded by*

$$Regret(T) \leq \tau UpperBound(Regret_0(T\backslash\tau)) + o(\tau).$$

Due to space limitation, we omit the proof of this conclusion.

IV. NUMERICAL RESULTS

We implement our proposed policy in Matlab and conduct simulations to evaluate its effectiveness.

In our scenario, there is a multi-hop wireless network, where N flows with the interesting traffic passing through L links. The relationship between links and flows are generated randomly. Packet transmission of each flow is drawn from a distinct i.i.d Bernoulli stochastic process with a mean $p_f \in (0,1)$. The traffic on all the links along a path has the same statistics. The monitoring system can perform initial detection in its serving area, and identify the L links as its monitoring objects.Capture probability matrix is randomly generate $\boldsymbol{P_c} \in (0,1]$. In the simulations, we vary network topology (including the number of monitored link L and interesting flow N) and the scale of the monitoring system (i.e., the number of sniffers S). The simulation with each setting is repeated for 100 times to even random disturbance.

Fig. 1 shows some representative results for the regret of our policy over time, where $N = 5, S = 4, T = 100000$. We can observe that the regret tends to flatten out over time. The similar results can be found under various settings. It indicates the convergence of Algorithm 1 and verifies its effectiveness. We further vary $\tau = 0, 5, 10$ to study the impact of the communication period τ on the regret of the algorithm. We find that the resulted regret becomes larger with the increasing of τ and it is up to an additive penalty depending on τ.

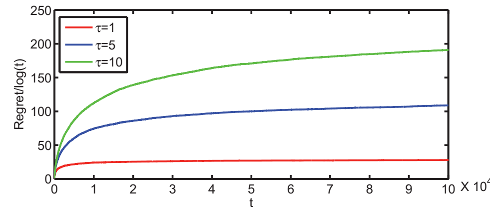

Fig. 1. The regret of Algorithm 1 with varying τ when $N = 5, S = 4$

V. CONCLUSION

In this paper, we investigate sniffer-channel assignment problem for data capture in multi-hop multi-channel wireless networks. We consider unreliable monitoring conditions and focus on sniffer redundancy deployment that utilized to improve capture probability. Without assuming any prior knowledge of the flow traffic, the sniffer-channel assignment is formulated as a combinatorial MAB problem, and a cooperative distribute learning policy is proposed.We analyze the regret of our policy in theory and simulation results validate that our policy can achieve logarithmic regret in the number of time slots. In the future work, we will extend our solution to other scenarios where the flows follow a Markovian process or under an adversarial setting.

ACKNOWLEDGMENT

This work was partially supported by National Natural Science Foundation of China under Grant No. 61301127, 61371080 and 61300223, the Fundamental Research Funds for the Central Universities, HUST (2015QN085), the US NSF under Grant No. CNS-1502584 and CNS-1464487,.

REFERENCES

[1] J. Yeo, M. Youssef, and A. Agrawala, "A framework for wireless lan monitoring and its applications," in *Proc. IEEE WISE'2004*, Singapore, Oct. 2004, pp. 70–79.

[2] D.-H. Shin and S. Bagchi, "Optimal monitoring in multi-channel multi-radio wireless mesh networks," in *Proc. ACM MOBIHOC'2009*, U.S.A, May 2009, pp. 229–238.

[3] N. H. S. G. Chhetri, A and Z. R., "On quality of monitoring for multi-channel wireless infrastructure networks," in *Proc. ACM MOBI-HOC'2010*, U.S.A, Sep. 2010, pp. 111–120.

[4] Q. Yan, M. Li, F. Chen *et al.*, "Non-parametric passive traffic monitoring in cognitive radio networks," in *Proc. IEEE INFOCOM'2013*, Italy, Apr. 2013, pp. 1240–1248.

[5] S. C. Arora, P and Z. R., "Sequential learning for optimal monitoring of multi-channel wireless networks," in *Proc. IEEE INFOCOM'2011*, China, Apr. 2011, pp. 1152–1160.

[6] J. Xu, Q. Wang, K. Zeng, M. Liu, and W. Liu, "Sniffer channel assignment with imperfect monitoring for cognitive radio networks," *IEEE Trans. on Wireless Commun.*, vol. 15, no. 3, pp. 1703–1715, 2016.

[7] N. Cesa-Bianchi and G. Lugosi, "Combinatorial bandits," *Journal of Computer and System Sciences*, vol. 78, no. 5, pp. 1404–1422, 2012.

[8] G. K. Audhya, K. Sinha, S. C. Ghosh, and B. P. Sinha, "A survey on the channel assignment problem in wireless networks," *Wireless Comm. and Mobile Comput.*, vol. 11, no. 5, pp. 583–609, 2011.

[9] Y. Gai, B. Krishnamachari, and R. Jain, "Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations," *IEEE Trans. on Netw.*, vol. 20, no. 5, pp. 1466–1478, 2012.

[10] Y. J. A. Zhang, L. Qian, and J. Huang, "Monotonic optimization in communication and networking systems," *Found. Trends Netw.*, vol. 7, no. 1, pp. 1–75, 2013.