Distributed Online Channel Assignment Toward Optimal Monitoring in Multi-Channel Wireless Networks

Dong-Hoon Shin, Saurabh Bagchi, and Chih-Chun Wang
School of Electrical and Computer Engineering, Purdue University
465 Northwestern Avenue, West Lafayette, IN 47907, USA
Email: {shin39, sbagchi, chihw}@purdue.edu

***Abstract*—This paper studies an optimal channel assignment problem for passive monitoring in multi-channel wireless networks, where a set of sniffers capture and analyze the network traffic to monitor the network. The objective of this problem is to maximize the total amount of traffic captured by sniffers by judiciously assigning the radios of sniffers to a set of channels. This problem is NP-hard, with the computational complexity growing exponentially with the number of sniffers. We develop *distributed online* solutions to this problem for large-scale and dynamic networks. Prior works have attained a constant factor $1 - \frac{1}{e}$ of the maximum monitoring coverage in a centralized setting. Our algorithm preserves the same ratio while providing a distributed solution that is amenable to online implementation. Also, our algorithm is cost-effective, in terms of communication and computational overheads, due to the use of only local communication and the adaptation to incremental network changes. We present two operational modes of our algorithm for two types of networks that have different rates of network changes. One is a proactive mode for fast varying networks, while the other is a reactive mode for slowly varying networks. Simulation results demonstrate the effectiveness of the two modes of our algorithm.**

I. Introduction

We consider a channel assignment problem for passive monitoring in multi-channel wireless networks. Passive monitoring is a widely-used and effective technique to monitor wireless networks, where a set of sniffers (i.e., software or hardware devices that intercept and log packets) are used to capture and analyze network traffic between other nodes to estimate network conditions and performance. Such estimates are utilized for efficient network operation, such as network resource management, network configuration, fault detection/diagnosis and network intrusion detection. Recently, it has been extensively studied to use multiple channels in wireless networks, especially in wireless mesh networks (WMNs) [1]–[6]. It has been shown that equipping nodes with multiple radios tuned to different non-overlapping channels can significantly increase the capacity of the network. In multi-channel wireless networks, a major challenge with passive monitoring is how to assign a set of channels to each sniffer's radios such that as large an amount of traffic or large a number of nodes as possible are captured. We call this the *optimal sniffer-channel assignment* (OSCA) problem.

Previous works [7]–[9] have studied variants of OSCA in different perspectives. In our prior work [7], we have studied a problem of how to optimally place sniffers and assign their channels to monitor multi-channel WMNs, assuming stationary networks. Chhetri *et al.* [8] have studied two models of sniffers that assume different capabilities of sniffers' capturing traffic. The first, called *user-centric* model, assumes that frame-level information can be captured so that activities of different users are distinguishable. The second, called *sniffer-centric* model, assumes only binary information regarding channel activities, i.e., whether some user is active in a specific channel near a sniffer. In both of the works [7], [8], the authors assume that a prior knowledge of the topology and the channel usages of nodes to be monitored is given to, or can be inferred by, sniffers. On the other hand, Arora *et al.* [9] have studied a trade-off between assigning the radios of sniffers to channels known to be busiest based on the current knowledge, versus exploring channels that are under observed. In addition, Subhadrabandhu *et al.* [10]–[12] have studied a problem of how to optimally place a set of intrusion detection modules for misuse detection in *single-channel* wireless networks.

One can obtain a good approximate solution to OSCA, which is an NP-hard problem (see Section II-B), by extending the algorithms in [7], [8]. The work [8] studies a special case of OSCA, where each sniffer has only one radio, while our prior work [7] studies a generalized version of OSCA, i.e., the optimal selection of sniffers and their channels. But, the algorithms in [7], [8] are centralized and offline algorithms. That is, the algorithms require a central authority that first gathers, from all sniffers, either a prior knowledge of the network topology and the channel usages of all nodes to be monitored [7], or primitive information to estimate the prior knowledge [8], then runs the algorithm and distributes the solution to all sniffers.

These centralized algorithms are not suitable for large-scale and dynamic networks, due to several reasons. The centralized algorithms require an efficient and cost-effective two-way global communication mechanism between the central authority and all sniffers, i.e., the communications from all sniffers to the central authority for the delivery of the prior knowledge, and also the communication from the central authority to all sniffers for the distribution of the solution. However, this is difficult to achieve in large-scale networks, especially in multi-hop wireless networks. Also, such a two-

978-1-4673-0775-8/12/$31.00 ©2012 IEEE

way global communication needs to be achieved without too much delay, otherwise the centralized algorithms are not agile to frequent network changes, such as channel-usage changes of nodes and network topology changes due to mobility of nodes and arrivals/departures of sniffers. In addition, the centralized algorithms are difficult to be deployed in ad hoc wireless networks, which lack the central authority or a powerful node that has a high computational power, a large memory, and no significant energy constraint. Moreover, the powerful node needs to be fault-tolerant or easily replaceable when it fails, since otherwise the entire monitoring system may fail due to a single-point failure.

In this paper, we develop *distributed* and *online* solutions to OSCA for large-scale and dynamic networks. Our distributed algorithm, called DA-OSCA, can always achieve at least $1-\frac{1}{e}$ (≈ 0.632) of the maximum monitoring coverage, regardless of the network topology and the channel assignment of nodes to be monitored. Previously, the centralized algorithms in [7], [8] have also attained the ratio $1-\frac{1}{e}$. However, our DA-OSCA preserves the same ratio while providing a distributed solution that is amenable to online implementation. Also, DA-OSCA is *cost-effective*, in terms of communication and computational overheads, since it requires only local communication among neighboring nodes and also adapts incrementally to network changes. Moreover, the decentralized and adaptive structure of DA-OSCA allows us to implement DA-OSCA in two operational modes; one is a proactive mode for fast-varying networks, while the other is a reactive mode for slow-varying networks. We demonstrate through simulations the effectiveness of the two modes of DA-OSCA.

II. Problem Formulation

A. Optimal Sniffer-Channel Assignment (OSCA) Problem

We are given a set N of nodes to be monitored, and each node $n \in N$ is tuned to a wireless channel chosen from a set C of available wireless channels, where $|C| \geq 2$. Channels are chosen according to one of many available channel assignment algorithms (e.g., [3], [4], [13]). Each node n is given a non-negative weight w_n. These weights of nodes can be used to capture various application-specific objectives of monitoring. For example, one can assign higher weights to the nodes that transmit larger volumes of data, thereby biasing our algorithm to monitor such nodes more. Or, for security monitoring, one can assign higher weights to more suspicious nodes.

We are given a set S of sniffers, each of which needs to determine a wireless channel from C to tune its radio to. We say that a sniffer and a node are *neighbors* if the sniffer can overhear the node, and also that two sniffers are *neighbors* if there exists a node that can be overheard by both the sniffers. We say that a node is *covered* if the node is overheard by at least one sniffer being tuned to the same channel as the node. We are given a collection of coverage-sets, $\mathcal{K} = \{K_{s,c} \subseteq N : s \in S,\, c \in C\}$, where a *coverage-set* $K_{s,c}$ contains the nodes that can be covered by sniffer s being tuned to channel c. We define a *group* as a collection of all coverage-sets of a sniffer, i.e., $\mathcal{K}_s = \{K_{s,c} : c \in C\}$. Our objective is to maximize the total weight of the nodes covered by judiciously choosing one coverage-set from each group. Here, the constraint that only one coverage-set can be chosen from each group arises since each sniffer has a single radio and can thus tune its radio to only one channel at a time. We refer to this problem as the ***optimal sniffer-channel assignment*** (OSCA) problem.

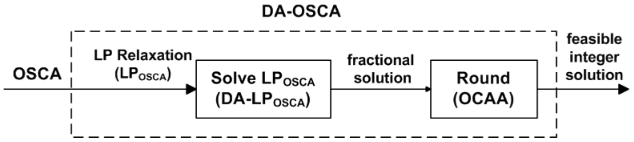


Fig. 1. Distributed Algorithm for solving OSCA (DA-OSCA).

For ease of exposition, we assume that all of the nodes and the sniffers have only one radio. However, the multi-radio case, where nodes and sniffers are equipped with multiple radios, can be easily mapped to this single-radio case by regarding radios of a node (or a sniffer) as different nodes (or sniffers) with a single radio.

B. Hardness of OSCA

We present existing results on the hardness of OSCA.

***Theorem** 1 (Theorem 1 [8]):* OSCA is NP-hard.

This means that the computational complexity to solve OSCA grows exponentially with the number of sniffers, unless $P = NP$.

Also, we have an inapproximability result for OSCA.

***Theorem** 2 (Corollary 2 [8]):* For any $\epsilon > 0$, it is NP-hard to approximate OSCA within a factor of $\frac{7}{8}+\epsilon$ of the optimum.

Thus, the best achievable approximation ratio for OSCA is at most $\frac{7}{8}$.

III. The Distributed Algorithm for OSCA

We develop a distributed algorithm to solve OSCA, referred to as DA-OSCA. The basic structure of DA-OSCA is based on the Linear Program (LP) rounding technique. In LP rounding, we first solve the LP relaxation of OSCA, and then round the (fractional) solution of the LP relaxation to an integer solution that is feasible to the original OSCA problem. Figure 1 shows an overview of how DA-OSCA yields an approximate solution to OSCA. DA-OSCA consists of two components: 1) Distributed Algorithm for solving the LP relaxation of OSCA (DA-LP$_{\text{OSCA}}$); 2) Opportunistic Channel Assignment Algorithm (OCAA) which performs distributed rounding of the fractional solution yielded by DA-LP$_{\text{OSCA}}$.

A. Distributed Algorithm for Solving LP relaxation of OSCA

LP relaxation of OSCA. We first formulate OSCA into an integer linear program (ILP). We assign an indicator variable $x_n \in \{0,1\}$ to each node $n \in N$, where $x_n = 1$ indicates that node n is covered by the given solution. We assign an indicator variable $y_{s,c} \in \{0,1\}$ to a coverage-set $K_{s,c} \in \mathcal{K}$, and $y_{s,c} = 1$ indicates that sniffer s will be tuned to channel c. The ILP of OSCA, denoted by ILP$_{\text{OSCA}}$, is given by

ILP$_{\text{OSCA}}$:

$$
\begin{aligned}
\text{maximize} \quad & \sum_{n\in N} w_n x_n && && (1)\\
\text{subject to} \quad & x_n \le \sum_{s,c:\, n\in K_{s,c}} y_{s,c} && \forall n \in N, && (2)\\
& \sum_{c\in C} y_{s,c} \le 1 && \forall s \in S, && (3)\\
& 0 \le x_n, y_{s,c} \le 1 && \forall n \in N, s \in S, c \in C, && (4)\\
& x_n, y_{s,c} \in \{0,1\} && \forall n \in N, s \in S, c \in C. && (5)
\end{aligned}
$$

Since ILP$_{\text{OSCA}}$ cannot be solved in polynomial time, we relax the integer constraint (5) and obtain the LP relaxation of OSCA, i.e., Eqs. (1)–(4), denoted by LP$_{\text{OSCA}}$. In LP$_{\text{OSCA}}$, the variables x_n's and $y_{s,c}$'s can now take any value in $[0,1]$, including fractional values.

DA-LP$_{\text{OSCA}}$. We present the Distributed Algorithm for solving LP$_{\text{OSCA}}$ (DA-LP$_{\text{OSCA}}$) in Alg. 1. DA-LP$_{\text{OSCA}}$ is based on the *Proximal Optimization Algorithm* (POA) [14, Ch. 3.4.3]. The derivation of DA-LP$_{\text{OSCA}}$ is provided in [15].

DA-LP$_{\text{OSCA}}$ has two levels of iterations: inner-level iterations (i.e., the `for` loop in lines 2–7) and outer-level iterations (i.e., the `while` loop in lines 1–9). In the outer-level iterations, DA-LP$_{\text{OSCA}}$ sequentially updates the values of the two kinds of variables, i.e., first the primal variables x_n's and $y_{s,c}$'s, and then the auxiliary variables x_n^{aux}'s and $y_{s,c}^{\text{aux}}$'s. In the inner-level iterations, DA-LP$_{\text{OSCA}}$ updates the values of the primal variables through $I+1$ iterations, each of alternately updating the values of the primal variables and the dual variables p_n's. DA-LP$_{\text{OSCA}}$ can start with any initial values. In Alg. 1, d and β are positive constants. In Eq. (6), the projection $[\cdot]^+_{Y_s}$ can be easily obtained, e.g., using an algorithm provided in [15].

Note that DA-LP$_{\text{OSCA}}$ requires *only local communications* among neighboring nodes. In many monitoring applications, it would be desirable that DA-LP$_{\text{OSCA}}$ should be run by only sniffers since DA-LP$_{\text{OSCA}}$ is for sniffers to determine their channels. In such cases, we can let one of neighboring sniffers of node n act as a proxy that takes over the node n's duty of updating values of its variables x_n, x_n^{aux} and p_n. Thus, in the cases, each sniffer s needs to update values of its own variables $\vec{y}_s$ and $\vec{y}_s^{\text{aux}}$, and also variables x_n's, x_n^{aux}'s, and p_n's for the nodes that it acts as a proxy for.

The standard POA [14, Ch. 3.4.3], which is the DA-LP$_{\text{OSCA}}$ when $I \to \infty$, requires a two-level convergence structure. That is, the inner-level iterations must converge before the next outer-level iteration begins. However, such a two-level convergence structure is not suitable for distributed algorithms since it incurs substantial overheads due to a mechanism required to determine when to stop inner-level iterations. Hence, we fix the number of inner-level iterations of DA-LP$_{\text{OSCA}}$ to 2 (i.e. $I = 1$), and find a good approximate solution.

Algorithm 1 DA-LP$_{\text{OSCA}}$

```
1: while TRUE do
2:     for $i = 0$ to $I$ do
3:         Each node $n$ computes $x_n(i)$ according to:
               $$x_n(i) \leftarrow [x_n^{\text{aux}} + d(w_n - p_n(i))]^+_{[0,1]}.$$
           Also, each sniffer $s$ computes $\vec{y}_s(i)$ according to:
               $$\vec{y}_s(i) \leftarrow \left[\left(y_{s,c}^{\text{aux}} + d\sum_{n\in K_{s,c}} p_n(i) : c \in C\right)\right]^+_{Y_s}. \quad (6)$$
           Here, $Y_s = \{\vec{y}_s : \sum_{c\in C} y_{s,c} \le 1,\, y_{s,c} \ge 0\ \forall c\}$, and
           $[\vec{x}]^+_A$ denotes the projection to a set $A$. Then, sniffer
           $s$ sends the updated values $\vec{y}_s(i)$ to its neighboring nodes.
4:         if $i \ne I$ then
5:             Each node $n$ computes $p_n(i+1)$ according to:
                   $$p_n(i+1) \leftarrow [p_n(i) + \beta \cdot g_n(i)]^+_{[0,+\infty)}, \text{where}$$
                   $$g_n(i) \leftarrow x_n(i) - \sum_{(s,c):n\in K_{s,c}} y_{s,c}(i).$$
               Then, node $n$ sends $p_n(i+1)$ to its neighboring nodes and sniffers.
6:         end if
7:     end for
8:     Each node $n$ and each sniffer $s$ set initial values of their
       variables for the next iteration according to:
           $x_n^{\text{aux}} \leftarrow x_n(I)$ and $p_n(0) \leftarrow p_n(I)$      (node $n$)
           $\vec{y}_s^{\text{aux}} \leftarrow \vec{y}_s(I)$      (sniffer $s$).
9: end while
```

We now show that, even with $I = 1$, DA-LP$_{\text{OSCA}}$ can converge to the optimal solution. Let $\vec{x}^{\text{aux},t}$, $\vec{y}^{\text{aux},t}$, and $\vec{p}^t$ be the values of $\vec{x}^{\text{aux}}(I)$, $\vec{y}^{\text{aux}}(I)$, and $\vec{p}(I)$, respectively, at the t-th outer-level iteration of DA-LP$_{\text{OSCA}}$. Also, we let $(\vec{x}^*, \vec{y}^*)$ and $\vec{p}^*$ be the primal optimal solution and the dual optimal solution, respectively. We show the following theorem which provides a sufficient condition of β for the convergence of DA-LP$_{\text{OSCA}}$ with $I = 1$. The proof is provided in [15].

***Theorem** 3:* As $t \to \infty$, a sequence of vectors $(\vec{x}^{\text{aux},t}, \vec{y}^{\text{aux},t}, \vec{p}^t)$ given by DA-LP$_{\text{OSCA}}$ with $I = 1$ converges to $(\vec{x}^*, \vec{y}^*, \vec{p}^*)$, provided that

$$\beta < \frac{1}{2dB_1B_2},$$

where $B_1 = \max\{|K_{s,c}| : s \in S, c \in C\} + 1$, $B_2 = \max\{|C|, M+1\}$, and $M = \max_{n\in N} |\{K_{s,c} : n \in K_{s,c}\}|$.

Here, the upper bound $\frac{1}{2dB_2B_2}$ can be obtained by computing the two pieces of information: the maximum number of nodes that can be covered by any sniffer operating on any channel, and the maximum number of neighboring sniffers that a normal node has.

B. Opportunistic Channel Assignment Algorithm

We present a distributed rounding algorithm in Alg. 2, referred to as the *Opportunistic Channel Assignment Algorithm*

Algorithm 2 Opportunistic Channel Assignment Algorithm

```
// Assume a partition $\mathcal{P} = \{P_i\}$ of the set $S$ of all sniffers
    such that no two sniffers in any $P_i$ are neighbors.
for $i = 1$ to $|\mathcal{P}|$ do
    // All sniffers in $P_i$ can choose their channels in parallel.
    Each sniffer $s \in P_i$ tunes its radio to a channel $c^* \in C$
    such that
        $I(K_{s,c^*}; \vec{y}^*_{N(s)}) = \max_{c \in C} I(K_{s,c}; \vec{y}^*_{N(s)})$, where
        $I(K_{s,c}; \vec{y}^*_{N(s)}) = \sum_{n \in K_{s,c}} w_n \prod_{(s',c): s' \neq s, n \in K_{s',c}} (1 - y^*_{s',c})$,
    and $N(s)$ denotes the set of neighboring sniffers of
    sniffer $s$.
    After determining its channel, the sniffer $s$ sends the
    determination to its neighboring sniffers.
end for
```

(OCAA). It determines the channel assignment of sniffers based on the optimal solution $\vec{y}^*$ given by DA-LP$_{\text{OSCA}}$. In OCAA, a novel metric $I(K_{s,c}; \vec{y}^*_{N(s)})$ is introduced in order to guide each sniffer to make a good decision on selecting its channel. We can interpret $I(K_{s,c}; \vec{y}^*_{N(s)})$ as the expected coverage improvement that sniffer s can achieve by tuning its radio to channel c, by viewing $y^*_{s',c}$ as the probability that sniffer s' tunes its radio to channel c.

OCAA has the following performance guarantee.

***Theorem** 4:* Given an approximate solution to LP$_{\text{OSCA}}$ that attains a constant factor α of the optimal value of LP$_{\text{OSCA}}$, OCAA guarantees to achieve at least $\alpha \cdot (1 - \frac{1}{e})$ ($\approx 0.632\alpha$) of the maximum monitoring coverage of OSCA.

The proof is provided in [15]. Here, note that we can make the approximate solution to LP$_{\text{OSCA}}$ arbitrarily close to the optimal solution as we increase the number of outer-level iterations of DA-LP$_{\text{OSCA}}$. Hence, DA-OSCA can always achieve at least $1 - \frac{1}{e}$ of the maximum monitoring coverage of OSCA.

IV. Online Implementation of DA-OSCA

In this section, we discuss how DA-OSCA can be implemented for online operation so that DA-OSCA is agile and adapts incrementally to network changes. We discuss two operational modes of DA-OSCA that are suitable for fast-varying and slow-varying networks, respectively.

A. Proactive mode of DA-OSCA for fast-varying networks

For fast-varying networks, we implement DA-OSCA to operate *proactively* so that it can quickly adapt to frequent network changes. In this proactive mode, DA-OSCA executes one outer-level iteration of DA-LP$_{\text{OSCA}}$ every T_1 time units, and invokes OCAA every lT_1 time units. That is, DA-OSCA keeps updating the primal and the dual variables (using DA-LP$_{\text{OSCA}}$), and periodically changes the channel assignment of sniffers (using OCAA) based on the updated values of $\vec{y}$.

B. Reactive mode of DA-OSCA for slow-varying networks

For slow-varying networks, we implement DA-OSCA to operate *on demand*, i.e., only when it needs to change the channel assignment of sniffers to improve the degraded monitoring coverage. For this reactive operational mode, we need a mechanism to evaluate the quality of monitoring coverage in order to determine when to start and also when to terminate DA-OSCA. Hence, we first present a procedure to evaluate the quality of monitoring coverage, and then discuss how we can implement DA-OSCA to operate in a reactive mode using this procedure.

To evaluate the quality of monitoring coverage, sniffers perform a sequential procedure along a pre-constructed spanning tree of them. The procedure is initiated by leaf sniffers and is executed sequentially along the levels of the spanning tree upwards to the root sniffer. At a level of the spanning tree, sniffer s computes the followings:

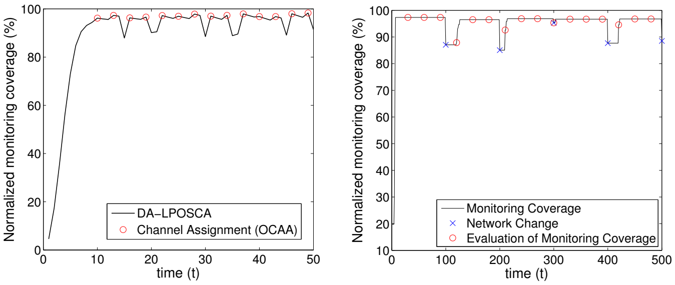
$$C_s = \sum_{s' \in \text{CS}(s)} C_{s'} + \sum_{n \in L(s)} w_n \cdot \min\Big\{1, \sum_{(s,c): n \in K_{s,c}} y_{s,c}\Big\},$$
$$D_s = \sum_{s' \in \text{CS}(s)} D_{s'} + \sum_{n \in K_{s,c^*}} p_n + \sum_{n \in L(s)} [w_n - p_n]^+, \qquad (7)$$

where $c^* \in \text{argmax}_{c \in C} \sum_{n \in K_{s,c}} p_n$, $[x]^+ = \max\{x, 0\}$, and CS(s) and $L(s)$ denote the set of the child sniffers of sniffer s and the set of neighboring nodes of sniffer s, respectively. Finally, the root sniffer computes C_{root} and D_{root} according to Eq. (7), and then decides to start DA-OSCA, or to terminate DA-LP$_{\text{OSCA}}$, by checking if $C_{\text{root}} \geq \gamma \cdot D_{\text{root}}$, where γ is a predetermined threshold. (If the condition is met, it is guaranteed that the current monitoring coverage achieves at least γ of the maximum coverage [15].) Then, the determination is delivered to all sniffers along the spanning tree.

We now describe how DA-OSCA can be implemented to operate in a reactive mode using the above procedure. In this mode, DA-OSCA evaluates the quality of the current monitoring coverage periodically, e.g., every T_2 time units, by employing the above procedure. If the quality of the current monitoring coverage is above a desired level (in terms of γ), DA-OSCA terminates doing nothing. Otherwise, DA-OSCA starts to run outer-level iterations of DA-LP$_{\text{OSCA}}$ to improve the degraded monitoring coverage. At every several outer-level iterations of DA-LP$_{\text{OSCA}}$, DA-OSCA checks whether the current solution of DA-LP$_{\text{OSCA}}$ is sufficiently close to the optimal solution of LP$_{\text{OSCA}}$ by using the above procedure. Once a near-optimal solution to LP$_{\text{OSCA}}$ is obtained, DA-OSCA terminates DA-LP$_{\text{OSCA}}$ and then rounds the solution of LP$_{\text{OSCA}}$ with OCAA to obtain an feasible integer solution. Then, DA-OSCA terminates.

V. Simulation

We conduct simulations to demonstrate the efficacy of the two modes of DA-OSCA. In the simulation, 500 nodes of identical weight and 50 sniffers are randomly deployed in the network with a uniform distribution. The number of available wireless channels is three (i.e., $|C| = 3$). The



(a) Proactive mode

(b) Reactive mode

Fig. 2. Evolution of monitoring coverage in two modes of DA-OSCA.

channel of each node is assigned randomly to channel 1, 2, or, 3 with probabilities 0.2, 0.3, and 0.5, respectively. The channel assignment of a fraction of nodes (randomly chosen between 10% and 40%) changes every 5 time units and every 100 time units in the fast-varying and slow-varying networks, respectively. Here, we define one time unit as the time that DA-OSCA takes to run one outer-level iteration of DA-LP$_{\text{OSCA}}$.

Figure 2(a) shows how the monitoring coverage evolves as DA-OSCA in Mode-I adapts to fast channel changes of nodes. The monitoring coverage is normalized by the optimal value of LP$_{\text{OSCA}}$, which is an upper bound on the maximum monitoring coverage. In this experiment, DA-OSCA adjusts the channel assignment of sniffers after 10 time units after the experiment begins. We observe that the fractional monitoring coverage due to the solution of DA-LP$_{\text{OSCA}}$ converges rapidly (within 10 time units) until it reaches about 90% of the maximum coverage, and it flattens out after it goes above 90% of the maximum coverage. We also observe that DA-LP$_{\text{OSCA}}$ quickly recovers the degraded fractional monitoring coverage. Within only a few time units, the new channel assignment of sniffers by OCAA attains a high monitoring coverage, maintained above 95% of the maximum coverage.

Figure 2(b) demonstrates the on-demand operation of DA-OSCA in Mode-II for slow channel changes. We see large intervals of time where the monitoring coverage is flat. This means that DA-OSCA determined that the monitoring coverage meets the desired level through the procedure that evaluates the monitoring coverage (in Section IV-B), and then terminates without any processing, thereby saving unnecessary cost. We notice that when the network changes, the monitoring coverage suffers (note the dips) but quickly recovers (always within 20 time units) as OCAA is executed on demand.

Both experiments demonstrate that DA-OSCA is able to adapt to different kinds of networks, fast-varying and slow-varying, and is able to operate incrementally with respect to network changes. By setting the values of threshold (i.e., γ), the system operator can control how close she wants the normalized monitoring coverage to get to the value of one.

VI. Conclusion

In this paper, we presented a distributed online algorithm for the optimal channel assignment problem for passive monitoring in multi-channel wireless networks. Our algorithm preserves the approximation ratio $1 - \frac{1}{e}$ that the existing centralized algorithms have previously attained, while providing a distributed solution that is amenable to online implementation. We discuss two operational modes of our algorithm for cost-effective operation in two types of networks that have different rates of network changes. Simulation results demonstrate the effectiveness of the two modes of our algorithm.

Our future work is on how to make our distributed algorithm execute asynchronously. Further, we are studying the security monitoring where a node needs to covered by multiple sniffers for reliable monitoring, due to imperfect sniffers.

VII. Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant Nos. CNS-0831999 and ECCS-0925851. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] M. Alicherry, R. Bhatia, and L. Li, “Joint Channel Assignment and Routing for Throughput Optimization in Multi-radio Wireless Mesh Networks,” in *Proc. of ACM MobiCom*, 2005.

[2] P. Kyasanur, J. So, C. Chereddi, and N. H. Vaidya, “Multi-Channel Mesh Networks: Challenges and Protocols,” *IEEE Wireless Communications*, April 2006.

[3] A. Raniwala and T. Chiueh, “Architecture and Algorithms for an IEEE 802.11-Based Multi-Channel Wireless Mesh Network,” in *Proc. of IEEE INFOCOM*, March 2005.

[4] X. Lin and S. Rasool, “A Distributed Joint Channel-Assignment, Scheduling and Routing Algorithm for Multi-Channel Ad Hoc Wireless Networks,” in *Proc. of IEEE INFOCOM*, 2007.

[5] V. Bhandari and N. H. Vaidya, “Capacity of Multi-Channel Wireless Networks with Random (c, f) Assignment,” in *Proc. of ACM MobiHoc*, September 2007.

[6] A. Dhananjay, H. Zhang, J. Li, and L. Subramanian, “Practical, Distributed Channel Assignment and Routing in Dual-radio Mesh Networks,” in *Proc. of ACM SIGCOMM*, August 2009.

[7] D.-H. Shin and S. Bagchi, “Optimal Monitoring in Multi-Channel Multi-Radio Wireless Mesh Networks,” in *Proc. of ACM MobiHoc*, May 2009.

[8] A. Chhetri, H. Nguyen, G. Scalosub, and R. Zheng, “On quality of monitoring for multi-channel wireless infrastructure networks,” in *Proc. of ACM MobiHoc*, September 2010.

[9] P. Arora, C. Szepesvari, and R. Zheng, “Sequential Learning for Optimal Monitoring of Multi-channel Wireless Networks,” in *Proc. of IEEE INFOCOM*, April 2011.

[10] D. Subhadrabandhu, S. Sarkar, and F. Anjum, “A Framework for Misuse Detection in Ad Hoc Networks—Part I,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 274–289, February 2006.

[11] ——, “A Framework for Misuse Detection in Ad Hoc Networks—Part II,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 290–304, February 2006.

[12] ——, “A Statistical Framework for Intrusion Detection in Ad Hoc Networks,” in *Proc. of IEEE INFOCOM*, April 2006.

[13] P. Kyasanur and N. H. Vaidya, “Routing and Link-layer Protocols for Multi-Channel Multi-Interface Ad Hoc Wireless Networks,” *SIGMOBILE Mobile Computing and Communications Review*, vol. 10, no. 1, pp. 31–43, January 2006.

[14] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Prentice-Hall, New Jersey, 1989.

[15] D.-H. Shin, S. Bagchi, and C.-C. Wang, “Distributed Online Channel Assignment Toward Optimal Monitoring in Multi-Channel Wireless Networks,” Purdue University, http://docs.lib.purdue.edu/ecetr/427, Tech. Rep., 2011.