Sniffer Channel Selection for Monitoring Wireless LANs

Yuan Song[1], Xian Chen[1], Yoo-Ah Kim[1], Bing Wang[1], and Guanling Chen[2]

[1] University of Connecticut, Storrs, CT 06269
[2] University of Massachusetts, Lowell, MA, 01854

Abstract. Wireless sniffers are often used to monitor APs in wireless LANs (WLANs) for network management, fault detection, traffic characterization, and optimizing deployment. It is cost effective to deploy single-radio sniffers that can monitor multiple nearby APs. However, since nearby APs often operate on orthogonal channels, a sniffer needs to switch among multiple channels to monitor its nearby APs. In this paper, we formulate and solve two optimization problems on sniffer channel selection. Both problems require that each AP be monitored by at least one sniffer. In addition, one optimization problem requires minimizing the maximum number of channels that a sniffer listens to, and the other requires minimizing the total number of channels that the sniffers listen to. We propose a novel LP-relaxation based algorithm, and two simple greedy heuristics for the above two optimization problems. Through simulation, we demonstrate that all the algorithms are effective in achieving their optimization goals, and the LP-based algorithm outperforms the greedy heuristics.

1 Introduction

Wireless LANs (WLANs) have been widely deployed in enterprise and campus networks. A number of studies use air sniffing as an effective technique for understanding and monitoring WLANs [11, 6, 1, 7, 2, 8]. In air sniffing, sniffers are placed inside a WLAN, each passively listening to the air waves in its vicinity, and collecting detailed MAC/PHY information. This detailed low-level information provides valuable insights into the behavior of wireless medium and protocols. It is also critical for effective monitoring and management of WLANs.

Large-scale WLAN monitoring through air sniffing, however, faces several challenges. First, it requires a large number of sniffers, which can be costly to deploy and difficult to manage. This problem is compounded by the fact that APs in WLANs can operate on different channels (e.g., 802.11b/g supports 3 orthogonal channels, and 802.11a supports 13 orthogonal channels), while an air sniffer only listens to a single channel at any point of time (multi-radio sniffers are large and expensive to deploy [3]). Therefore, in the worst case, the required number of sniffers can be the same as the number of APs. Secondly, the sniffers generate a large amount of measurement data, which can be expensive to store, transfer and process. For instance, in [2], up to 80 Mbps of traffic is generated for monitoring an academic building, which needs to be transferred and processed at a central server.

To overcome the above challenges in large-scale air sniffing, [3] proposes *channel sampling*, where each sniffer samples the network traffic by visiting multiple channels periodically. Using channel sampling, a sniffer can monitor multiple nearby APs that operate on different channels, and hence less sniffers are needed. Furthermore, the sampling of traffic leads to less amount of measurement data. As shown in [3], channel sampling is useful for a number of applications, including security monitoring, anomaly detection, fault diagnosis, network characterization, and assistance to AP deployment. The study of [3] proposes two sampling strategies, equal-time sampling where a sniffer spends equal amount of time scanning each channel, and proportional sampling where the amount of time that a sniffer spends on a channel is proportional to the amount of traffic on that channel. These two strategies are improved in [4] where the scanning of the sniffers are coordinated to increase the number of unique frames.

In this paper, we address an important problem in channel sampling, namely, how to select the channels for the sniffers. Our study differs from [3, 4] in two main aspects. First, we require each sniffer to monitor a *subset* of selected channels so that each AP is monitored by at least one sniffer, while [3, 4] require each sniffer to monitor all available channels (regardless of whether the channels are being used or not by the nearby APs). By eliminating the scanning over unused channels, our approach provides more effective traffic sampling[3]. Secondly, we formulate and solve two optimization problems on sniffer channel selection: one minimizes the maximum number of channels that a sniffer listens to, and the other minimizes the total number of channels that the sniffers listen to[4]. The first objective is desirable because when a sniffer monitors less channels, it can spend more time on each of these channels; the second objective is desirable because it may reduce the number of sniffers needed (it may need less sniffers than the first objective, see Section 4). We develop a novel LP-relaxation based algorithm, and two simple greedy heuristic algorithms for the above two optimization problems. Through simulation, we demonstrate that all the algorithms are effective in achieving their optimization goals, and the LP-based algorithm outperforms the greedy heuristics.

The rest of the paper is organized as follows. Section 2 describes the problem setting. Sections 3 and 4 describe our sniffer channel selection algorithms and their evaluation, respectively. Finally, Section 5 concludes the paper.

2 Problem setting

Consider a WLAN with a set of APs, V. Each AP uses a single radio, and hence a single channel, at any point of time (if an AP uses multiple channels simultaneously, we can regard it as multiple APs, each with a single channel). Let C denote the set of channels that the APs operate on. In particular, suppose AP v operates on channel c_v, $c_v \in C$. A set of sniffers, M, is spread out in the

[3] One motivation of scanning all the channels in [3, 4] is that it can capture rogue APs that operate on unused channels. Rogue APs, however, can be effectively detected using other approaches such as [9, 10].

[4] In practice, a network administrator may choose one of these two objectives based on the goals of the WLAN monitoring.

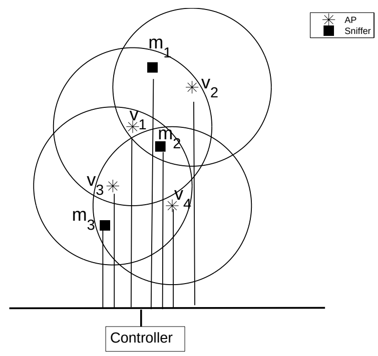


Fig. 1. Problem setting: a central controller controls the channels of the APs and determines the channel assignment for the sniffers.

WLAN to monitor the APs. Let R_v denote the set of sniffers that are within the transmission range of v (i.e., R_v is the set of sniffers that can overhear the transmission of v when listening to channel c_v), $R_v \subseteq M$. We assume that $|R_v| \geq 1$, i.e., at least one sniffer can monitor v, $\forall v \in V$. Each sniffer has a single radio, and switches among multiple channels to monitor its nearby APs when these APs operate on different channels.

We assume that the WLAN uses a centralized management architecture (as commonly used in commercial products), where a central controller manages the operation of the APs. We assume that the central controller knows the location of the APs, and determines the channel for each AP. Furthermore, it knows the location of the sniffers, and determines the set of channels that each sniffer scans based on the locations of the APs, sniffers, and the channels of the APs. Fig. 1 illustrates the centralized management architecture. In this example, four APs, v_1, v_2, v_3, v_4, and three sniffers, m_1, m_2, m_3, are controlled by the centralized controller. APs v_1 and v_3 operate on channel 1; APs v_2 and v_4 operate on channel 2. Sniffer m_1 is in the transmission ranges of v_1 and v_2; sniffer m_2 is in the transmission ranges of all four APs; sniffer m_3 is in the transmission ranges of v_3 and v_4.

Let $\varphi(v)$ denote the set of sniffers that monitor AP v, referred to as *assignment* to v. Correspondingly, let $C_\varphi(m)$ denote the set of channels that sniffer m monitors. Then $C_\varphi(m) = \{c_v \mid m \in \varphi(v)\}$. We look at two variants of sniffer channel selection. Both variants require that each AP be monitored by at least one sniffer, i.e., $\varphi(v) \neq \emptyset, \forall v \in V$. In addition, the first variant requires minimizing the maximum number of channels that a sniffer listens to, i.e., minimizing $\max_{m \in M} |C_\varphi(m)|$. The second variant requires minimizing the sum of the channels that the sniffers listen to, i.e., minimizing $\sum_{m \in M} |C_\varphi(m)|$. We refer to these two variants as *min-max* and *min-sum sniffer channel selection problems*, respectively.

3 Algorithms for sniffer channel selection

In this section, we develop three algorithms for sniffer channel selection. The first algorithm is based on LP relaxation. The second and third algorithms both use a greedy heuristic, targeting at the min-max and min-sum problem, respectively. We refer to the two greedy heuristics as *Greedy-max* and *Greedy-sum*, respectively.

3.1 LP-relaxation based algorithm

The main idea of this algorithm is as follows. We first formulate the sniffer channel selection problem (the min-max or min-sum problem) as an integer programming (IP) problem, and then solve its corresponding linear programming (LP) problem (by relaxing the integer constraints). After obtaining the optimal solution to the LP problem, we convert it to the integer solution to the original IP problem. More specifically, let $x_{m,c}$ be a 0-1 random variable, $x_{m,c} = 1$ denotes that sniffer m monitors channel c; and $x_{m,c} = 0$ denotes otherwise. Then the min-max sniffer channel selection problem can be formulated as:

$$\text{minimize}: \max_{m \in M} \sum_{c \in C} x_{m,c} \tag{1}$$

$$\text{subject to:} \sum_{m \in R_v} x_{m,c_v} \geq 1, \forall v \in V \tag{2}$$

$$x_{m,c} \in \{0, 1\} \tag{3}$$

Similarly, the min-sum problem can be formulated as an IP problem by simply replacing the objective function (1) with

$$\text{minimize}: \sum_{m \in M} \sum_{c \in C} x_{m,c} \tag{4}$$

We relax the integer constraint on $x_{m,c}$, and let $y_{m,c} \in [0, 1]$ be the relaxed value of $x_{m,c}$. The original IP problems then become LP problems, which can be solved easily. After solving for $y_{m,c} \in [0, 1]$, consider an AP $v \in V$ and the values of y_{m,c_v} for $m \in R_v$ (i.e., the sniffers in the transmission range of v). We round $y_{m,c}$ to obtain x_{m,c_v} as follows. Since an AP only needs to be monitored by one sniffer, we choose one monitor, m', that is closest to 1 among $y_{m,c_v}, \forall m \in R_v$ (i.e., it satisfies $y_{m',c_v} = \max_{m \in R_v} y_{m,c_v}$), and set x_{m',c_v} to 1.

Algorithm 1 summarizes this LP-relaxation based algorithm. Line 1 solves the LP problem (for the min-max or min-sum objective function) to obtain $y_{m,c} \in [0, 1]$, $\forall m \in M, \forall c \in C$. Line 2 initializes $x_{m,c}$ to zero, $\forall m \in M, \forall c \in C$. The algorithm then considers all the APs. For an AP $v \in V$, if one monitor, $m \in R_v$, is already assigned to monitor v's channel, c_v, we simply assign m to monitor v; otherwise, we set x_{m',c_v} to 1 where m' satisfies that $y_{m',c_v} = \max_{m \in R_v} y_{m,c_v}$. In the rest of the paper, when this LP-relaxation based algorithm solves the

Algorithm 1 LP-relaxation based sniffer channel assignment

Solve the LP program (objective function (1) for min-max problem; objective function (4) for min-sum problem) to obtain $y_{m,c}, \forall m \in M, \forall c \in C$
$x_{m,c} = 0, \forall m \in M, \forall c \in C$.
$C_\varphi(m) = \emptyset$
for all $v \in V$ **do**
  if $\exists m \in R_v$ s.t. $x_{m,c_v} = 1$ **then**
    $\varphi(v) = \{m\}$
  else
    $m' = \arg\max_{m \in R_v} y_{m,c_v}$
    $x_{m',c_v} = 1$
    $C_{m'} = C_{m'} \cup \{c_v\}$
    $\varphi(v) = \{m'\}$
  end if
end for
Return(φ, C_φ)

min-max LP problem, we refer to it as *LP-max*; otherwise (i.e., it solves the min-sum LP problem), we refer to it as *LP-sum*.

We now illustrate this LP-relaxation based algorithm using the example in Fig. 1. Solving the min-max LP problem, we have $y_{m_1,1} = 0, y_{m_1,2} = 1, y_{m_2,1} = 1, y_{m_2,2} = 0, y_{m_3,1} = 0$, and $y_{m_3,2} = 2$. After the LP rounding, we have $C_\varphi(m_1) = \{2\}$, $C_\varphi(m_2) = \{1\}$, and $C_\varphi(m_3) = \{2\}$, leading to a solution of 1 for the min-max problem. Solving the min-sum LP problem, we have $y_{m_1,1} = y_{m_1,2} = 0, y_{m_2,1} = y_{m_2,2} = 1$, and $y_{m_3,1} = y_{m_3,2} = 0$. After the LP rounding, we have $C_\varphi(m_1) = \emptyset$, $C_\varphi(m_2) = \{1, 2\}$, and $C_\varphi(m_3) = \emptyset$, leading to a solution of 2 for the min-sum problem.

We now present an approximation-ratio result for the LP-relaxation based algorithm.

Theorem 1 *LP-max is an $O(r)$-approximation algorithm for the min-max sniffer channel selection problem, and LP-sum is an $O(r)$-approximation algorithm for the min-sum problem, where $r = \max_{v \in V} |R_v|$, i.e., r is the maximum number of sniffers that are in the transmission range of an AP.*

Proof. Consider an arbitrary AP, v, and a sniffer $m \in R_v$. Our LP rounding guarantees that $x_{m,c_v} \le r y_{m,c_v}$. This can be shown by considering the following two cases. When $y_{m,c_v} = \max_{m \in R_v} y_{m,c_v}$, by our LP rounding, $x_{m,c_v} = 1$, and we have $x_{m,c_v} \le r y_{m,c_v}$ (since $y_{m,c_v} \ge 1/r$). When $y_{m,c_v} \ne \max_{m \in R_v} y_{m,c_v}$, by our LP rounding, $x_{m,c_v} = 0 \le r y_{m,c_v}$. Since the above AP, v, is chosen arbitrarily, we have

$$\sum_{c \in C} x_{m,c} \le r \sum_{c \in C} y_{m,c}, \forall m \in M.$$

Let n_m^* represent the optimal solution to the min-max sniffer channel selection problem. We have

$$\max_{m \in M} \sum_{c \in C} x_{m,c} \le r(\max_{m \in M} \sum_{c \in C} y_{m,c}) \le r n_m^*. \tag{5}$$

The second inequality above is because the LP provides a lower bound to the original problem. From (5), LP-max is an $O(r)$-approximation algorithm for the min-max sniffer channel selection problem. Similarly, let n_s^* represent the optimal solution to the min-sum problem. We have

$$\sum_{m \in M} \sum_{c \in C} x_{m,c} \leq r\left(\sum_{m \in M} \sum_{c \in C} y_{m,c}\right) \leq r n_s^*. \tag{6}$$

Hence LP-sum is an $O(r)$-approximation algorithm for the min-sum problem.

3.2 Greedy-max algorithm

Algorithm 2 Greedy-max

$\varphi(v) = \{m \mid m \in R_v\}, \forall v \in V$
$C_\varphi(m) = \emptyset$, $V_{m,c} = \emptyset$, $\forall m \in M, c \in C$
for all $v \in V$ **do**
  for all $m \in M$ **do**
    if $m \in R_v$ **then**
      $C_\varphi(m) = C_\varphi(m) \cup \{c_v\}$, $V_{m,c_v} = V_{m,c_v} \cup \{v\}$
    end if
  end for
end for
repeat
  $M' = \emptyset$
  for all $m \in M$ **do**
    if $\exists c \in C_\varphi(m)$ s.t. $\forall v \in V_{m,c}, |\varphi(v)| \geq 2$ **then**
      $M' = M' \cup m$
    end if
  end for
  if $M' \neq \emptyset$ **then**
    Let m be a monitor in M' that monitors the largest number of channels
    $C_\varphi(m)' = \{c \mid c \in C_\varphi(m), |\varphi(v)| \geq 2, \forall v \in V_{m,c}\}$
    Pick $c \in C_\varphi(m)'$ that has the smallest $|V_{m,c}|$
    $C_\varphi(m) = C_\varphi(m) \setminus \{c\}$
    $\varphi(v) = \varphi(v) \setminus \{m\}, \forall v \in V_{m,c}$
    $V_{m,c} = \emptyset$
  end if
until M' is empty
Return(φ, C_φ)

Greedy-max heuristic is designed for the min-max objective. Its main idea is as follows. Initially, a sniffer, m, is assigned to monitor an AP, v, as long as m is in the transmission range of v. The algorithm then runs in iterations. In each iteration, it finds the sniffer with the maximum number of channels and removes one channel from this sniffer when feasible (i.e., while still satisfying the monitoring constraints). The iteration stops when none of the sniffers can remove any channel.

Algorithm 2 summarizes this algorithm. Line 1 initializes $\varphi(v)$ to be the set of sniffers that are in the transmission range of v, $\forall v \in V$. Let $V_{m,c}$ denote the set of APs that sniffer m monitors on channel c. Lines 2-9 initialize $C_\varphi(m)$ and $V_{m,c}$, $\forall m \in M, c \in C$. In each iteration (lines 11-23), let M' record the set of sniffers that can remove at least one channel. It then picks a sniffer, m, that monitors the maximum number of channels from M'. Afterwards, it finds a channel, c, that can be removed and removes it from $C_\varphi(m)$ (if multiple such channels exist, it chooses to remove the channel with the smallest number of APs, see line 20). Last, line 22 removes m from the assignment of all the APs in $V_{m,c}$ (since m does not monitor channel c any more).

When using this algorithm to solve the example in Fig. 1, we have $C_\varphi(m_1) = \{2\}$, $C_\varphi(m_2) = \{1\}$, $C_\varphi(m_3) = \{2\}$, leading to a solution of 1 for the min-max problem, and a solution of 3 for the min-sum problem.

3.3 Greedy-sum algorithm

Algorithm 3 Greedy-sum

$C_\varphi(m) = \emptyset$, $V_{m,c} = \emptyset$, $\forall m \in M, c \in C$
for all $v \in V$ **do**
  for all $m \in M$ **do**
    if $m \in R_v$ **then**
      $V_{m,c_v} = V_{m,c_v} \cup \{v\}$
    end if
  end for
end for
$V' = V$
repeat
  pick m, c such that $|V_{m,c}| = \max_{m' \in M, c' \in C} |V_{m',c'}|$
  $\varphi(v) = \{m\}, \forall v \in V_{m,c}$
  $C_\varphi(m) = C_\varphi(m) \cup \{c\}$
  $V_{m',c} = V_{m',c} \setminus V_{m,c}, \forall m' \in M$
  $V' = V' \setminus V_{m,c}$
until V' is empty
Return(φ, C_φ)

Greedy-sum heuristic is designed for the min-sum objective. It models the sniffer channel selection problem as a minimum set covering problem: we map each sniffer to $|C|$ *virtual sniffers*, each monitoring one channel in C, then the min-sum problem is equivalent to finding the minimum number of virtual sniffers so that all APs are monitored and the number of virtual sniffers (and hence the sum of the channels used by all the sniffers) is minimized. Many algorithms have been proposed for the minimum set covering problem. Greedy-sum follows a greedy algorithm for minimum set covering problem [5]. It runs in iterations.

In each iteration, it picks a sniffer and channel pair that monitors the maximum number of APs. The iteration stops when all the APs are monitored.

Algorithm 3 summarizes this algorithm (we used a similar algorithm for scheduling sniffers to detect rogue APs in [10]). Let $V_{m,c}$ denote the set of APs that sniffer m monitors on channel c. Line 1 initializes $C_\varphi(m)$ to be an empty set, and lines 1-8 initialize $V_{m,c}$, $\forall m \in M, c \in C$. Line 9 initializes, V', the set of APs that are not monitored by a sniffer, to V. The algorithm then run in iterations until V' is empty. Using a greedy strategy, line 11 chooses the monitor, m, and the channel, c, such that $|V_{m,c}| = \max_{m' \in M, c' \in C} |V_{m',c'}|$ (if multiple such sniffers exist, we choose the one with the minimum $|C_\varphi(m)|$). Line 12 assigns m to all the APs in $V_{m,c}$; and line 13 adds channel, c, into $C_\varphi(m)$. Afterwards, since the APs in $V_{m,c}$ have already been monitored, line 14 removes $V_{m,c}$ from $V_{m',c}, \forall m' \in M$, and line 15 removes $V_{m,c}$ from V'.

Following the results in [5], the approximation ratio of Greedy-sum is H_d for the min-sum problem, where $H_d = \sum_{i=1}^{d} 1/i$ is the d-th harmonic number, and d is the maximum number of APs that a sniffer can monitor in its neighborhood.

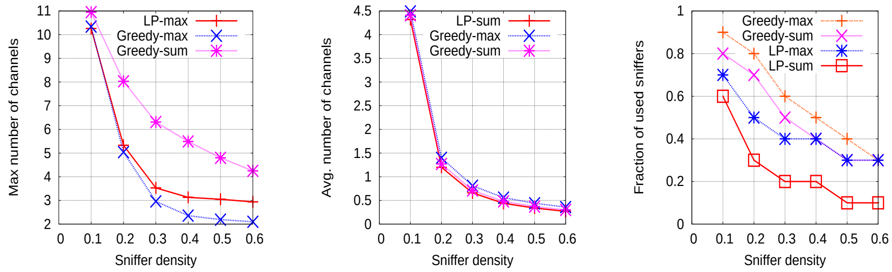
When using this algorithm to solve the example in Fig. 1, we have $C_\varphi(m_1) = \emptyset$, $C_\varphi(m_2) = \{1, 2\}$, $C_\varphi(m_3) = \emptyset$, leading to a solution of 2 for the min-sum problem, and a solution of 2 for the min-max problem.

4 Performance evaluation

Our performance evaluation uses an empirical dataset that contains both the coordinates and channels for the APs deployed at Dartmouth campus. We consider two 500 m $\times$ 500 m areas in this data set: one has approximately 400 APs, and the other has approximately 200 APs. These APs use both 802.11b/g and 802.11a, and operate on 12 orthogonal 2.4GHz/5GHz channels. The transmission range of each AP is set to 100 m.

To systematically evaluate the performance of our algorithms, for each area we consider, we generate $10,000$ topologies by virtually placing sniffers uniformly randomly into the area. The number of sniffers is randomly chosen from 1 to the number of APs. For each topology, we obtain a pair (n_a, n_s), where n_a is the number of APs that can be monitored by at least one sniffer, and n_s is the number of sniffers that can monitor at least one AP (i.e., sniffers that within the transmission range of at least one AP). Therefore n_a and n_s can be smaller than the number of APs and sniffers in the area, respectively). We refer to the ratio, n_s/n_a, as *sniffer density*.

Fig. 2 shows the results for the area with 400 APs (the results for the area with 200 APs are similar). Fig. 2(a) plots the maximum number of channels that a sniffer monitors. The results of the two algorithms that target at this optimization goal (i.e., LP-max and Greedy-max) are plotted in the figure. For comparison, we also plot the results under Greedy-sum. The x-axis of the figure represents sniffer density, i.e., n_s/n_a. The results are aggregated over a bin size of 0.1, i.e., the result under $n_s/n_a = x$ is the average of all the topologies with $n_s/n_a \in (x - 0.1, x]$. We observe that for all three algorithms, as expected, the maximum number of channels used by the sniffers reduces as the sniffer



(a) (b) (c)

Fig. 2. Results for 400 APs: (a) maximum number of channels that a sniffer monitors, (b) average number of channels that a sniffer monitors, and (c) fraction of sniffers that are used for monitoring.

density increases. Furthermore, Greedy-max slightly outperforms LP-max, and both Greedy-max and LP-max outperform Greedy-sum. We also observe a diminishing gain from increasing the density of sniffers: the maximum number of channels decreases dramatically first and then less dramatically afterwards. Fig. 2(b) plots the average number of channels that a sniffer monitors. The results of the two algorithms that target at this optimization goal (i.e., LP-sum and Greedy-sum) are plotted in the figure. For comparison, we also plot the results under Greedy-max. We observe that all three algorithms lead to similar performance, and LP-sum slightly outperforms the other two. Again, we observe a diminishing gain from increasing the density of sniffers.

For all the algorithms, the channel assignment solution may not assign a sniffer to monitor any AP (even though the sniffer is in the transmission range of some APs and can be used to monitor these APs). Fig. 2(c) plots the fraction of sniffers that are used (i.e., monitor at least one AP) for various sniffer densities under all the four algorithms. We observe that, for the same sniffer density, the LP-based algorithms require less sniffers than the greedy heuristics: LP-sum requires significantly less sniffers than Greedy-sum, and LP-max requires significantly less sniffers than Greedy-max (particularly for low sniffer densities). We also observe that the min-sum problem tends to require less sniffers than the min-max problem (e.g., the fraction of used sniffers is the lowest under LP-sum, much lower than that under LP-max). This is not very surprising since the channel assignment in the min-max problem needs to be balanced (for the min-max goal), while the min-sum problem does not have this requirement.

Last, combining the results in Figures 2(a),(b), and (c), we conclude that the LP-based algorithms outperform the two greedy heuristics since for both the min-max and min-sum problems, LP-max and LP-sum achieve similar objective values as their greedy counterparts, while using much less sniffers. We also observe that, when deploying sniffers at appropriate positions, the LP-based algorithms only require a small number of sniffers to achieve most of the gains. For instance, for the min-max problem, LP-max leads to a maximum of 3 channels over all sniffers when the number of sniffers is only 12% of the number of APs (this can be seen from Figures 2(a) and (c), which show that when the sniffer density is 0.3, the maximum number of channels is 3, and only 40% of the sniffers are used). For the min-sum problem, LP-sum only requires the number of sniffers to be as low as 6% of the number of APs (this can be seen from Fig. 2(c): for a sniffer density of 0.1, 0.2, and 0.3, the fraction of used sniffers is around 0.6, 0.3 and 0.2, respectively).

5 Conclusions

In this paper, we studied sniffer channel selection for monitoring WLANs. In particular, we formulated min-max and min-sum sniffer channel selection problems, and proposed a novel LP-relaxation based algorithm, and two simple greedy heuristic algorithms to solve them. Through simulation, we demonstrated that all the algorithms are effective in achieving their optimization goals, and the LP-based algorithm outperforms the greedy heuristics.

Acknowledgement. This work was partially supported by NSF CAREER award 0746841, UConn Faculty Large Grant, and DHS award NBCH2050002 subcontracted through Dartmouth ISTS.

References

1. R. Chandra, J. Padhye, A. Wolman, and B. Zill. A location-based management system for enterprise wireless LANs. In *NSDI*, 2007.
2. Y.-C. Cheng, J. Bellardo, P. Benko, A. C. Snoeren, G. M. Voelker, and S. Savage. Jigsaw: Solving the puzzle of enterprise 802.11 analysis. In *Proc. of ACM SIGCOMM*, Pisa, Italy, September 2006.
3. U. Deshpande, T. Henderson, and D. Kotz. Channel sampling strategies for monitoring wireless networks. In *Proc. of the Second Workshop on Wireless Network Measurements*, Boston, MA, April 2006.
4. U. Deshpande, C. McDonald, and D. Kotz. Coordinated sampling to improve the efficiency of wireless network monitoring. In *ICON*, November 2007.
5. D. S. Hochbaum. Approximating covering and packing problems: set cover, vertex cover, independent set, and related problems. *D. S. Hochbaum editor, Approximation algorithms for NP-hard problems*, pages 94–143, 1996. PWS Publishing Co. Boston, MA, USA.
6. A. P. Jardosh, K. N. Ramachandran, and K. C. Almeroth. Understanding link-layer behavior in highly congested IEEE 802.11b wireless networks. In *E-WIND*, 2005.
7. R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan. Analyzing the MAC-level behavior of wireless networks in the wild. In *Proc. of ACM SIGCOMM*, 2006.
8. A. Sheth, C. Doerr, D. Grunwald, R. Han, and D. C. Sicker. MOJO: A distributed physical layer anomaly detection system for 802.11 WLANs. In *Proc. of ACM MobiSys*, pages 191–204, 2006.
9. W. Wei, K. Suh, B. Wang, Y. Gu, J. Kurose, and D. Towsley. Passive online rogue access point detection using sequential hypothesis testing with TCP ACK-pairs. In *IMC*, 2007.
10. B. Yan, G. Chen, J. Wang, and H. Yin. Robust detection of unauthorized wireless access points. *ACM/Springer Mobile Networks and Applications*, In Press.
11. J. Yeo, M. Youssef, and A. Agrawala. A framework for wireless LAN monitoring and its applications. In *WiSe*, 2004.