A Green Scheduler for Enterprise WLANs

Kwan-Wu Chin
School of Electrical, Computers, and Telecommunications Engineering,
University of Wollongong, Australia
kwanwu@uow.edu.au

***Abstract*—In light of worldwide concerns regarding the emission of green house gases, researchers have begun to look at ways to reduce the carbon footprint of Information Technology (IT) infrastructure. This paper contributes to this effort by proposing a scheduler that switches off unnecessary Access Points (APs), up to 88% of deployed APs, while ensuring end-users continue to enjoy good coverage and performance.**

I. Introduction

Enterprises are beginning to deploy redundant, centrally controlled Access Points (APs) to provide high bandwidth, mobility and reliability service to their workers. As pointed by the authors of [4], Intel Corporation has deployed 125 APs at its Oregon office, where each AP serves four to six cubicles. Another example is shown in Figure 1, where five APs are used to cover 23 stations. In practice, each AP will have a reduced transmission range and operates on an orthogonal channel, with each station associated to the AP with strongest signal strength. Note that given the high density of APs, stations are likely to be located near an AP and hence, are likely to to experience high data rates [5]. Moreover, as each AP only needs to support a fraction of the total number of stations, there is less contention as well as interference – both of which leads to better network capacity.

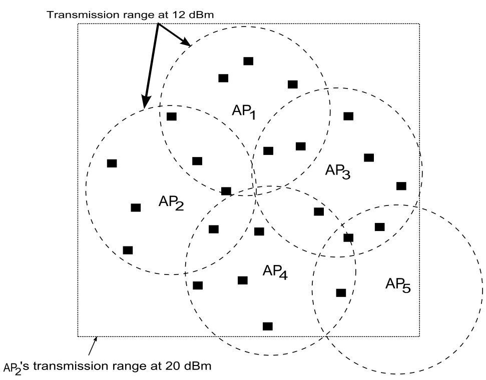

Fig. 1. A dense WLAN example. Filled squares denote stations.

The main cost of having dense Wireless Local Area Networks (WLANs) is the high power consumption. Interestingly, a past study [4] has shown that WLANs are for the most part under utilized, and exhibit diurnal patterns that are characterized by idle periods in the order of several hours. For example, 50% of the APs deployed at Dartmouth college's campus experience more than 60 minutes of idle times. In addition, the study shows 10-80% of APs at an Intel WLAN to be idle in a month. These observations have therefore motivated researchers to devise various ways to scale power usage in terms of user demands. That is, "green" schedulers that intelligently switch off non essential equipments in times of low usage, e.g., at night. In a prior work, Jardosh et al. [4] proposed a resource on demand strategy that switches off unnecessary APs. Their strategy addresses four main problems: clustering, demand estimation, topology management and user management. As we will show in Section VI, a key limitation of their clustering phase is that it does not consider the Quality of Service (QoS) of stations, and guarantee appropriate coverage.

This paper, therefore, addresses this limitation by proposing a centralized scheduler that switches APs on and off dynamically whilst maintaining coverage and performance. In terms of the WLAN shown in Figure 1, a naive scheduler may power off AP_1 and $AP_3 \dots AP_5$ in the evening, and increase the transmission power of AP_2 to 20 dBm to provide adequate coverage to all users. This, however, means stations that are near AP_5 will receive poor service as they are farther away from AP_2 as compared to AP_5. In this respect, our scheduler carefully balances the energy consumption of APs and QoS requirements of users.

In the next section, we first formulate the problem as a variant of the maximum coverage problem with budget constraints. Then, in Section III, we instantiated the formulated problem in the context of a WLAN. Section IV presents the proposed green scheduler, followed by our simulation methodology in Section V. We present our results in Section VI, and conclusions in Section VII.

II. The Problem

Our problem is similar to the minimum cost set covering problem. Define a ground set χ, and a family $\Upsilon = S_1, S_2, \dots, S_n$ of subsets of χ, where $S_i \subseteq \chi$. Moreover, each S_i has cost c_i. The well known set covering problem is to find a cardinality $J \subseteq \{1, \dots, n\}$ that covers all members of χ, i.e., $\cup_{j \in J} S_j = \chi$ and $\sum_{j \in J} c_j$ is minimum. For our problem, we further define $G = \{G_1, \dots, G_\ell\}$ to be subsets of Υ, with each G_i having a budget B_i. This additional formulation is equivalent to the set cover with group budget problem; see [1] for details on NP hardness and approximation bounds. Note that a key distinction to [1] is that there is no overall budget B as we assume APs have unlimited capacity to the Internet.

Our aim is to find the minimum number of APs that cover and maximize the utility of all stations. Specifically, find a solution $H \subseteq \{S_1, S_2, \dots, S_n\}$ such that $|H \cap G_i| \leq B_i$, and both $|H|$ and $c(H)$ are minimum. Here, the function $c(.)$ is defined as $\sum_{i \in H} c_i$. Once H is found, we switch off APs that satisfy the condition $H \cap G_i = \emptyset$.

III. System Model

The system under consideration consists of a set of APs that are managed by a central entity These APs are usually connected by one or more switches, which may also supply them with power via IEEE 802.3af – aka Power over Ethernet (PoE). Each AP draws 10W of power, whereas a switch with 24 to 72 ports consumes up to 350W per hour [4].

Each AP has access to p power levels; e.g., Cisco Aironet cards provide between 5-6 discrete power levels [6]. This implies that the set Υ is finite. Hence, in an enterprise with 100 APs, each with five power levels, we have $|\Upsilon| = 500$. We assume APs and stations support IEEE 802.11k [3], and thereby have the following functionalities: (i) APs broadcast a site report that is created using feedback containing APs that are within range and their signal strength from stations, (ii) APs are able to request stations to gather channel utilization and interference information from non-802.11 devices, (iii) APs and stations have the ability to collect performance data such as packet loss, signal quality and throughput, (iv) APs can be instructed to track hidden terminals or stations at the edge of their cell, and (v) both APs and stations are able to negotiate and adjust their transmission power. This last functionality is important as stations and APs can increase or reduce their transmit power when we switch on/off APs.

The ground set χ corresponds to stations in an enterprise WLAN, whereas Υ consists of subsets of stations covered by different APs at different transmission power. The set G_i corresponds to subsets of χ that are covered by AP_i. For example, if AP_1 has two sets of stations, e.g., $\{1, 2\}$ and $\{1, 2, 3, 4\}$, that are reachable via two different power levels respectively, then $G_1 = \{\{1, 2\}, \{1, 2, 3, 4\}\}$. The set Υ, and hence G_i, is constructed as follows. The central entity instructs APs to broadcast measurement frames in a round-robin manner at each power level. Stations are then requested, via Measurement Request element, to passively monitor these frames, and report back the observed signal strength. From these reports, the scheduler determines the stations that are reachable at different power levels from each AP, and also their nominal data rate. The reported data rate, r_z of a station z, is then used to calculate the cost c_i of S_i as follows:

$$T_i = \sum_{z \in S_i} r_z \tag{1}$$

where r_z is the average data rate of station z. c_i is then defined as $\frac{1}{log(T_i)}$. Instead of data rate, we could also use a station's signal strength. Alternatively, c_i can be defined as the total energy cost incurred by an AP when transmitting to stations in S_i.

Note that the set of stations, χ, is available readily as each station needs to associate and disassociate to/from their chosen AP, and hence, the scheduler has an accurate list of stations at all times. Other than that, it knows the list of APs and their location. Hence, the set G corresponds to the set of stations that are reachable via each AP, where $B_i = 1$ as each AP can only transmit using one power level at a time.

Lastly, the central controller switches off APs that are not in the set H, and configures the transmission power of those in H according to the chosen S_i. We assume stations support the upcoming IEEE 802.11v amendment to ensure smooth handover, which allows the central entity to inform stations to handoff to their new AP.

IV. Green Scheduler

We run our green scheduler, see Algorithm 1, until all users are covered by the sets in H. It starts by initializing the variable X' to the ground set χ. After that, in line $4 - 11$, it iterates through each group and picks the set S_j with the highest number of users covered and cost ratio. Similarly, in line-12, it determines the S_r with the highest said ratio, but amongst the sets retrieved from each group G_i. This set is then added to H, and the stations covered by S_r are then removed from X'. Note, in the case where stations have a minimum bandwidth requirement, stations will have to be removed from the corresponding S_i before running Algorithm 1.

input : Υ,χ
output: H
1 $H = \emptyset$;
2 $X' = \chi$;
3 **while** $X' \neq \emptyset$ **do**
4   **for** $i = 1, 2, \ldots, \ell$ **do**
5   **if** $c(H \cap G_i) \leq B_i$ **then**
6             $z \leftarrow \arg\max \frac{S_j \cap X'}{c(S_j)} \quad \forall S_j \in G_i$;
7             $A_i \leftarrow S_z$
8   **else**
9             $A_k \leftarrow \emptyset$
10
11   **end**
12     $r \leftarrow \arg\max \frac{A_i \cap X'}{c(A_i)}$;
13     $H \cup A_r$;
14     $X' = X' \setminus A_r$;
15 **end**
16 return H;

Algorithm 1: Proposed green scheduler.

The running time of the algorithm is $O(|\chi|p\ell)$ because the *while* loop in the worst case covers one station in each iteration – i.e., $|\chi|$ iterations. The *for* loop iterates through $|G| = \ell$ APs, where each AP has a maximum of p S_j sets corresponding to the stations covered at each power level.

V. Simulation Methodology

Our custom simulator consists of 50 APs and stations. Each AP is assumed to operate on a different channel. Both APs and stations are placed randomly in a $N \times N$ m^2 area, where N is varied from 400 to 1500. Moreover, they operate using IEEE 802.11a [2], and hence, support the following data rates: $r \in \{6, 9, 12, 18, 36, 48, 54\}$. Each of which has the following corresponding receiver sensitivity value (in dBm) $\{-87, -86, -86, -85, -80, -75, -71\}$. In accordance with [6], we choose four power levels (in dBm): $\{20, 17, 13, 10\}$. Each AP then transmits using these power levels to construct the sets S_j. The simulator uses the 2-ray ground model and the aforementioned receiver sensitivity values to determine whether a station is able to receive a packet.

We compare our scheduler to the AP clustering algorithm proposed by Jardosh et al. [4]. Briefly, each AP maintains a neighbor list and the number of neighbors. Their algorithm then picks the AP, say z, with the most neighbors as a cluster leader. After that, all neighbors that form a clique, including with AP-z, are added into the same cluster as AP-z. These APs are then removed from their respective neighbors' neighbor list. The algorithm then repeats until all APs are part of a cluster. Finally, their algorithm leaves cluster leaders powered on, whilst switches off all other group members. In our simulation, the leader is set to transmit at the highest power; i.e., 20 dBm.

We collect two main results: (i) the number of APs that needs to be powered on in order to support all stations, and (ii) the average utility of each station, which is defined as the average $log(r_z)$ value of all stations, where r_z is the average data rate of station z to the AP that transmitted a measurement frame. Lastly, we average the results over 100 simulation runs.

VI. Results

Figure 2 shows the number of APs that needs to be powered on for different values of N. Our green scheduler, on average, only powers on two to six APs; viz. 4% to 12% of the total APs. In comparison, Jardosh et al.'s algorithm power on 20% to 60% of the APs. The main reason is that Jardosh et al.'s algorithm does not consider station coverage directly. That is, it considers the proximity of APs as opposed to stations. In addition, only APs that form a clique are grouped into the same cluster. Our scheduler, however, only powers on the minimal number of APs that is required to cover all stations.

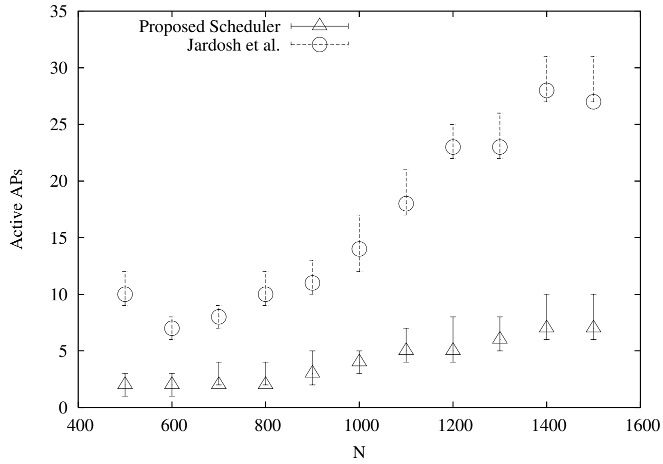

Fig. 2. Number of active APs with increasing N.

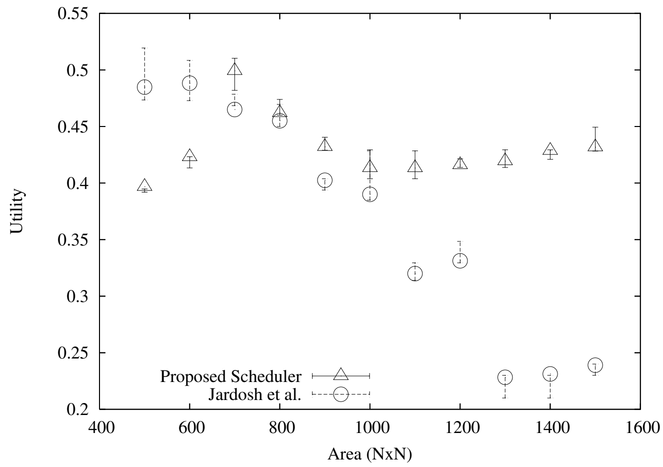

Fig. 3. The average utility of each station versus the sparseness of APs.

Figure 3 shows the utility received by each station. In a small area, Jardosh et al.'s algorithm provides better utility. The primarily reason is because there are more APs serving the given area; as illustrated in Figure 2. As a result, stations are more likely to obtain better data rate. However, as the placement of APs becomes sparser, stations receive poorer service because they are likely to be farther away from the cluster leader. On the other hand, our green scheduler takes stations' utility into account and therefore is able to strategically power on only those APs that provide good coverage to stations. In addition, unlike Jardosh et al's algorithm, our scheduler guarantees all stations are covered by an AP.

VII. Conclusion

We have outlined a green scheduler that computes the minimal number of APs required to service a set of stations by exploiting varying transmission powers whilst also taking into account the utility of stations. In particular, it is capable of powering off 50% more APs than the green clustering algorithm proposed in [4]. Hence, an interesting future work is to re-evaluate the resource on demand strategy of [4] using our scheduler on a WLAN test-bed.

References

[1] C. Chekuri and A. Kumar. Maximum coverage problem with group budget constraints and applications. In *Procs. of the seventh Intl. Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, Cambridge, MA, USA, Aug. 2004.

[2] IEEE. Part 11a: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: High speed physical layer in the 5 GHz band, standard specification. IEEE Std 802.11a-1999, 1999.

[3] IEEE. Amendment 1: Radio resource measurement of wireless LANs. IEEE Std 802.11k-2008, 2008.

[4] A. P. Jardosh, K. Papagiannaki, E. M. Belding, K. C. Almeroth, G. Iannaccone, and B. Vinnakota:. Green wlans: On-demand wlan infrastructures. *ACM/Spring MONET*, 14(6):798–814, June 2009.

[5] V. P. Mhatre and K. Papagiannaki. Optimal design of high density 802.11 WLANs. In *ACM CoNEXT*, Lisboa, Portugal, Aug. 2006.

[6] V. Shrivastava, D. Agrawal, A. Mishra, S. Banerjee, and T. Nadeem. Understanding the limitations of trnasmit power control for indoor WLANs. In *ACM IMC*, San Diego, CA, USA, Oct. 2007.