SNMP-Based Approach to Load Distribution in IEEE 802.11 Networks

Li-Hsing Yen
Dept. Computer Science & Information Engineering
Chung Hua University
Hsinchu, Taiwan 300, R.O.C.
Email: lhyen@chu.edu.tw

Tse-Tsung Yeh
Dept. Computer Science & Information Engineering
Chung Hua University
Hsinchu, Taiwan 300, R.O.C.
Email: nova.yeh@sti.com.tw

***Abstract*— In an IEEE 802.11 network, the traffic load collectively given by wireless stations (WSs) is usually not fairly shared by all available access points (APs), as WSs independently select APs to camp on. Prior approaches toward this problem either need to modify AP's behavior or require bandwidth negotiation and agreement enforcement between APs and WSs. These approaches are not practical due to their inability to apply to APs already in use. This paper proposes an application-layer approach, where a dedicated server is deployed to collects load-related information from APs utilizing SNMP (Simple Network Management Protocol). Our approach applies to off-the-shelf APs and has been proven very effective through thoughtful experiments.**

I. Introduction

IEEE 802.11 wireless local area networks (WLANs) have been widely deployed as an infrastructure providing wireless data access services in home, corporate, and public environments. A wireless station (WS) equipped with an 802.11 interface receives wireless connection services through an access point (AP). The WS must be associated with an AP before it can send packets.

The bandwidth offered by an AP is shared among all WSs associated with it. The throughput of an AP increases in proportional to the amount of packet traffic load added by all associated WSs, as long as the traffic load does not exceed the capacity of the AP. When the traffic load reaches some point, the AP's throughput may substantially degrade due to channel contention, an inherent problem of 802.11 medium access control (MAC) protocol. A simple way to increase overall system capacity is to deploy additional APs covering the same region, hoping that heavy traffic load can be distributed among these APs. Unfortunately, as each WS independently selects the AP to camp on, many WSs may be associated with the same AP while other APs remain idle. Consequently, the traffic load is not fairly shared by APs.

Prior work addressing this problem either needs to modify AP's behavior [1], [2], [3], [4] or requires WS to negotiate bandwidth allocation with the system and succeed in admission control before it can be associated with an AP [5]. These approaches are not practical, as conventional APs cannot benefit from these approaches. This paper aims to maximize network throughput by distributing WS's traffic load among APs. Unlike prior work, we take an application-layer approach that utilizes SNMP (Simple Network Management Protocol). A dedicated server uses SNMP protocol to periodically collects load-related information from APs. Upon entering a network, a WS contacts the server for the most appropriate AP that maximizes overall system throughput. The advantages of our work are summarized as follows.

- Our method applies to IEEE 802.11 as well as all the extensions (802.11 a/b/g).
- Any AP that supports SNMP can be applied without any modification.
- The firmware and driver of WS's wireless adaptor are left intact.

Accordingly, our proposal is more practical than counterparts. We have implemented our design with off-the-shelf APs and conducted experiments to measure its throughput improvement. The results show that this approach significantly increases overall throughput.

II. Problem Definition and Related Work

The association between a WS and an AP is created by the WS. Before a WS can send packets through an AP to the wired infrastructure, it should first discover available APs. To this end, the WS may perform either an active or a passive scan. In an active scan, the WS broadcasts in some channel a Probe Request message with a particular Service Set Identifier (SSID). If the SSID matches the one an AP is configured with, the AP responds a Probe Response to the WS, and the WS can therefore be aware of the presence of the AP. If the WS uses a passive scan instead, it does not issue any message but listens to Beacon messages broadcasted periodically by APs on channels of interest. With the AP information grabbed from the Probe Response or Beacon message, the WS selects a new AP to camp on based on the measure of received signal strength. Almost all existing 802.11 adaptors favor the AP with the strongest received signal strength.

IEEE 802.11 WLAN uses a contention-based MAC protocol. As a result, the overall system performance may degrade dramatically when too many WSs are associated with the same AP. Placing additional APs next to a crowded AP does not necessarily resolve the problem, as WSs may still choose the overloaded AP for its stronger signal strength.

Not much work has been proposed to address this issue. Some approaches [1], [2], [3] suggest that APs periodically broadcast load information (such as the number of associated WSs, the mean received signal strength, or packet error rate) so that WSs could select the AP that has the lightest load. These approaches need to modify AP's protocol stack as well as WS's. Velayos et al. [4] have proposed a distributed solution that requires APs to periodically calculate and exchange throughput (load) information. APs whose load exceeds the average by some value will be classified as overloaded and will transfer some WS to other under-loaded APs to maximize overall throughput. This method also needs to modify AP.

Balachandran et al. [5] assumed the deployment of a dedicated server that collects AP's load information. Upon entering a WLAN, a WS negotiates bandwidth allocation with the server. The server calculates available capacity of each AP and assigns the WS to one of the APs that meets the WS's bandwidth requirement. The goal is to evenly distribute traffic load among all APs. However, AP's available capacity is calculated based on the amount of allocated bandwidth, not bandwidth actually in use. As a result, the estimated capacity may not reflect the reality. Moreover, most APs lack an operational mechanism to enforce bandwidth agreement.

III. Proposed Scheme

We consider an Extended Service Set (ESS), where all APs are connected by Ethernet. Each AP must be configured with a valid IP address and run an SNMP agent software. A management workstation running an SNMP client program is assumed to located in the ESS, which we call an ALDP (Application-layer Load Distribution Protocol) server.

The proposed scheme deals with two issues. One is to detect the condition of overload (congestion) and the other is to alleviate the problem. They are discussed respectively in the following subsections.

A. Congestion Detection

Congestion refers to the condition that overall packet traffic contributed by all WSs exceeds the capacity of a single AP. When congestion occurs, packets will stay in sender's buffer much longer than the buffer can tolerate. Due to the dynamic nature of WLAN traffic, it is a challenge to accurately detect congestions.

In our design, the ALDP server periodically polls agent software running on APs to access statistics of packet traffic. The queried MIB (Management Information Base) objects (Table I) are all of MIB-II [6], which are common to networking devices such as switches and routers. IEEE802dot11-MIB [7], which is specific to 802.11 devices, is not used here. In most cases, only the difference of two successive polled values is of interest. We prefix Δ to a object name to represent the difference value.

From the polled values of these objects, the ALDP server calculates the utilization of an AP's wireless interface as follows.

$$\text{Utilization}\% = \frac{8 \times (\Delta\text{ifInOctets} + \Delta\text{ifOutOctets})}{\Delta\text{sysUptime} \times \text{ifSpeed}} \times 100. \quad (1)$$

TABLE I

MIB-II objects queried by ALDP server

Object name	Description
sysUptime	System up time in msec
ifInOctets	The count of inbound octets of traffic
ifOutOctets	The count of outbound octets of traffic
ifSpeed	The speed of the interface in bps
ifInErrors	The count of inbound packets
ifInUcastPkts	The count of inbound unicast packets
ifInNUcastPkts	The count of inbound non-unicast packets

This formula is suggested by Cisco [8]. One may conjecture that if we can gather a baseline for regular AP operations, an above-average value of utilization can be an indicator of congestion. However, we found through experiments that the utilization does not increase with the amount of traffic load; the utilization actually drops on congestions (see Sec. IV-A). Therefore, utilization alone cannot be used as an indication of congestion.

Another factor affecting traffic load is the number of WSs associated with an AP. Intuitively, a large number of WSs often implies high traffic load. However, this quantity alone can only be a rough estimate of overall traffic load as traffic conditions vary significantly among WSs. How to obtain this information is also a problem. Neither MIB-II [6] nor IEEE802dot11-MIB [7] has defined MIB object for the number of WSs currently associating with an AP. Although some vendors do have their own extensions to IEEE802dot11-MIB that include such information, these extensions only apply to their own products. We therefore decide to let ALDP server maintain such information at application level. Details will be presented in the next subsection.

Let U_i, N_i, and S_i be the utilization, the number of currently associated WSs, and the interface speed of AP i, respectively. For each AP i, a WS measures B_i which stands for the degree of benefit that can be derived if the WS is associated with AP i. B_i is given by

$$B_i = \frac{(100\% - U_i) \times S_i}{N_i + 1}. \quad (2)$$

Note that $(100\% - U_i) \times S_i$ is the residual bandwidth of AP i. It is normalized by $N_i + 1$ to obtain B_i, the normalized residual bandwidth (NRB) of AP i. Intuitively, B_i is high if both U_i and N_i are small. When U_i or N_i increases, B_i decreases accordingly. The WS will select the AP with the highest NRB to camp on.

B. Protocol Execution

Fig. 1 shows a message sequence chart for ALDP. The ALDP server periodically inquires APs for values of the objects defined in Table I (using SNMP Get-request/response messages). The ALDP server calculates residual bandwidth and maintains the number of associated WSs for each AP. When a WS has associated with an AP, it sends AP_STATUS_REQUEST to the ALDP server. The ALDP server replies AP_STATUS_RESPONSE, which carries the

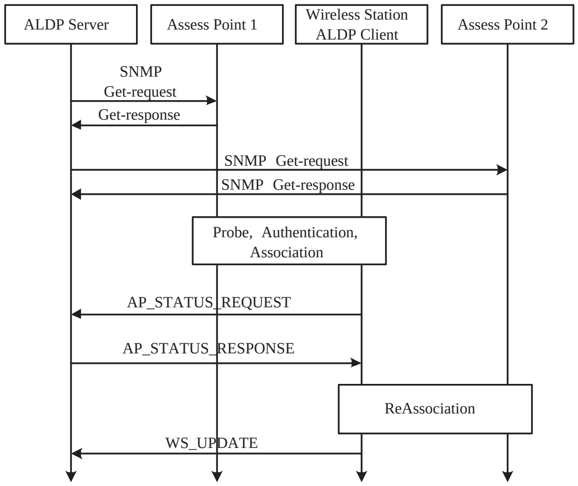

Fig. 1. ALDP message sequence chart

residual bandwidth and the number of associated WSs of every AP. Based on the returned information, the WS then selects the best AP by Eq. 2, and performs re-association if needed. After the re-association, the WS sends WS_UPDATE to the ALDP server to report its new choice of AP. This is for the ALDP server to maintain the up-to-date number of associated WSs in every AP.

In the above-mentioned scenario, the WS must learn of an ALDP server's IP address right after it has initially associated with some AP. This can be achieved by several means. In our scheme, the ALDP server in the current network domain is identified by a fully-qualified domain name (FQDN) that prefixes "aldp." to the current network domain name[1]. The WS is thus made aware of the IP address of its serving ALDP server through the current DNS (Domain Name Service) server. This approach allows the ALDP server to be installed in a network segment or virtual LAN different from where the WS resides. The ALDP server can also be installed in a DeMilitarized Zone (DMZ) that is protected by a firewall. This approach also has the merit of involving neither layer-2 nor layer-3 broadcasting.

C. Dynamic AP Selection

The basic ALDP scheme assumes that a WS selects a better AP to re-associate with right after it has associated with some AP but before it sends out any data packets to the network. Once the re-association has completed, the WS does not further switch to another AP. In some cases, allowing a working WS to dynamically change its AP can adapt to network dynamics and seems to be a good idea. However, to allow the dynamic AP-switching, the WS should be able to estimate the bandwidth it currently consumes. This is to quantify the traffic load that may shift from the current AP to the new AP in case of AP switching.

In the basic ALDP scheme, an ALDP client running on a WS inquires the ALDP server for the residual bandwidth and the number of WSs associated with each AP i, denoted by RB_i and N_i, respectively. To support dynamic AP selection, the WS also needs to run an SNMP agent. The ALDP client polls the agent software running on the same site for the utilization (calculated by Eq. (1)) and the data rate of its wireless interface. The bandwidth that the WS currently uses, denoted by *BW*, is given by the product of the utilization times the data rate of its wireless interface.

For each AP i that a WS is not currently associated, the predicted NRB of AP i if the WS switches to AP i is

$$B_i = \frac{RB_i - BW}{N_i + 1}. \tag{3}$$

Suppose the WS is currently associated with AP j. The predicted NRB of AP j if the WS leaves AP j is

$$B_j = \frac{RB_j + BW}{N_j - 1}. \tag{4}$$

The WS periodically calculates B_i's and will re-associate with AP i only if $B_i > B_j$ and B_i is the highest among all.

A potential problem with dynamic AP selection is ping-pong effect [2], which refers to the phenomenon of repeated AP switching from one AP to the other. The cause of ping-pong effect is due to uncoordinated but simultaneous AP switching among WSs. For example, if we suddenly boot an AP beside a congested AP, all WSs may detect the presence of the new AP and decide to switch to the new AP at almost the same time. Consequently, the new AP immediately becomes over-loaded due to the migration and all WSs decide to switch back. Then the same scenario repeats.

To avoid ping-pong effect, we introduce the notion of delay count (DC). Recall that a WS periodically searches for the best AP that has the highest NRB. When the best AP found is different from the previous one, the WS randomly generates a DC value for this AP. The WS can switch to this AP only after the AP has been identified as the best AP for *DC* successive times.

The range of *DC* has an impact on resultant performance. Let B_j and B_i be the NRBs of the current and the best AP, respectively. The range of DC value for AP i is given by

$$DC_i = \left\lceil \left(1 - \frac{B_i - B_j}{B_{max}}\right) \times 10 \right\rceil, \tag{5}$$

where B_{max} is the maximal possible value of AP i's NRB (occurs with zero utilization and $N_i = 1$). By definition, the DC value ranges from 0 to 10. In general, the DC value decreases as the differences between B_i and B_j increases.

IV. Experiments and Results

We implemented the proposed scheme and conducted experiments to measure its performance. In the experiments, we used two identical IEEE 802.11a APs (Cisco AIR-SP1220A), which support SNMP MIB-II, and placed them in a close range with a partition in between. They are referred to as AP 1 and AP 2, respectively. The APs were connected through Ethernet to an ALDP server running Linux and ALDP protocol. ALDP clients were implemented in notebook PCs, each was equipped

[1]For example, aldp.chu.edu.tw.

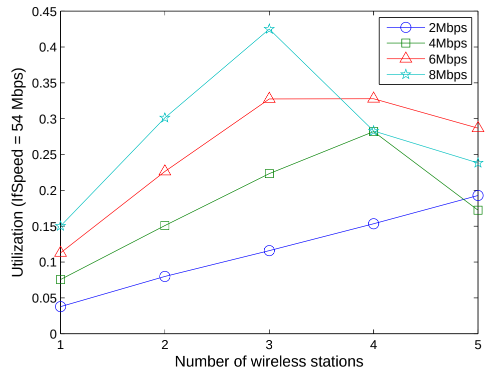

Fig. 2. Utilization vs. the number of associated WSs

with an IEEE 802.11 a/b/g PCMCIA interface card (D-Link DWL-AG660). The ALDP client ran Linux with MADWiFi[2] driver.

All throughput (utilization) measurements used data traffic generated by Iperf[3]. Iperf clients, each running at a WS, generated constant-rate UDP packets to an Iperf server, the ALDP server, where the measurements were obtained.

A. Utilization

We analyzed first the relationship between AP's utilization and the number of associated WSs for a single AP. The ALDP server measured AP's utilization by Eq. (1) every 10 seconds and the result was averaged over six measures. Fig. 2 shows the results, where the packet generation rate was varied 2 – 8 Mbps.

As can be seen from the figure, the utilization increases linearly with the amount of the traffic load collectively added by all WSs, as long as the load does not exceed AP's "real" capacity. The AP's real capacity is lower than the half of its theoretical capacity (54 Mbps) and decreases with the number of associated WSs. For example, consider two cases: (1) three WSs, each generated 8 Mbps data, and (2) four WSs, each generated 6 Mbps data. They offered the same traffic load, but the former had a higher utilization than that of the latter in Fig. 2. This can be explained as the real capacity is higher with three associated WSs than with four, which is reasonable since more WSs raise the degree of channel contention, reducing available bandwidth.

When the traffic load exceeded the real capacity, the utilization did not increase but started to drop with the number of WSs. We observed that WS's interface buffer was overrun at that time, an indication of network congestion. The results therefore confirm that the utilization does not increase with increased traffic load: the maximal utilization that can be achieved is limited by AP's real capacity.

[2]http://sourceforge.net/projects/madwifi/
[3]http://dast.nlanr.net/Projects/Iperf/

TABLE II

RECEIVED SIGNAL STRENGTHS AT EACH WS

	WS 1	WS 2	WS 3	WS 4
AP 1	-52 dBm	-48 dBm	-50 dBm	-58 dBm
AP 2	-60 dBm	-56 dBm	-62 dBm	-67 dBm

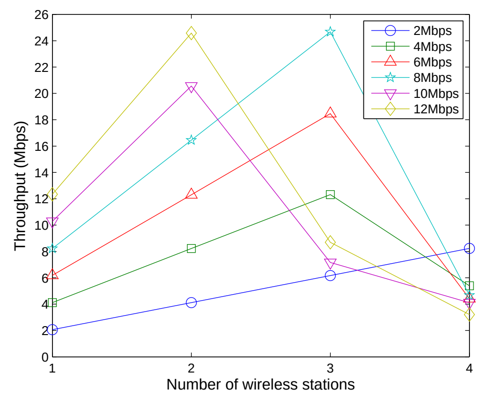

Fig. 3. Network throughput without ALDP

B. Performance of ALDP

We next measured the performance of ALDP in terms of network throughput. The first setup did not run ALDP. Four WSs were sequentially added to the network and placed in front of AP 1. Consequently, all WSs were associated with AP 1 for its stronger signal strength. Table II lists the received signal strengths reported at each WS. The network throughput in this case was contributed by AP 1 only. Fig. 3 shows the results.

The throughput increased linearly with the number of associated WSs when the traffic load did not exceed AP's real capacity. The throughput decreased when more WSs were added into the network. The system capacity was around 26 Mbps when the number of WSs was two or three and droped to 8 Mbps when four WSs participated.

We then investigated the effectiveness of ALDP. The setup is the same as the first one, except that ALDP was running.

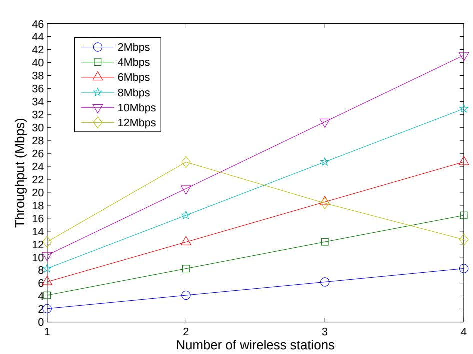

Fig. 4. Network throughput with ALDP

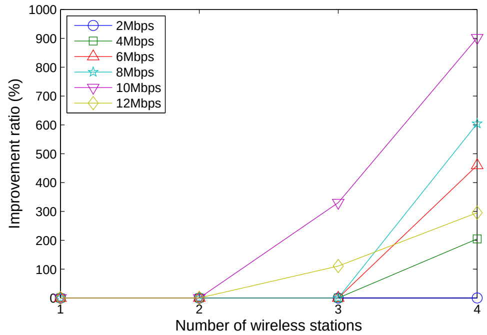

Fig. 5. Throughput improvement ratio of ALDP

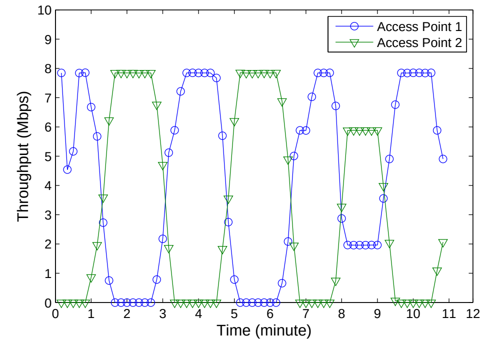

Fig. 6. The presence of ping-pong effects (DC = 1)

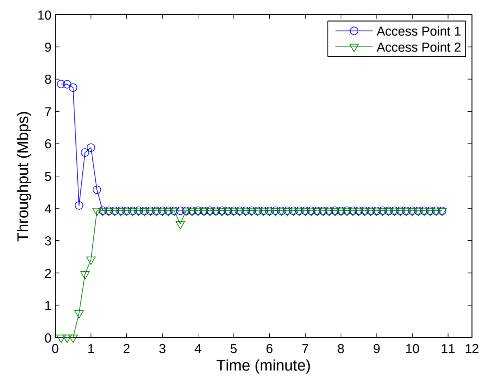

Fig. 7. The absence of ping-pong effects

WSs in this case were not always associated with AP 1, so the network throughput was contributed by both APs. The results are shown in Fig. 4.

We can see that, except for the case of 12 Mbps data traffic, the network throughput increases linearly with offered load. The maximal value reaches 40 Mbps. Clearly, ALDP effectively increases the system capacity. Fig. 5 shows throughput improvement ratio of ALDP, where the improvement ratio is defined as

$$\frac{\text{Throughput with ALDP} - \text{Throughput without ALDP}}{\text{Throughput without ALDP}}.$$

We can see that ALDP does not improve network throughput when traffic load does not exceed AP's capacity. When traffic load goes beyond AP's capacity yet is still within the system's capacity, ALDP significantly increases network throughput. The highest improvement ratio is 900%. When the offered load exceeds the system's capacity, ALDP still improves the overall performance to some extent.

C. Ping-Pong Effects

We ran experiments to observe ping-pong effects brought by dynamic AP selection. Four WSs were simultaneously turned on, each generating 2-Mbps UDP packets. Here we manually set the DC value to 1 so the design of delay count was effectively disabled. Fig. 6 shows how the throughput of AP 1 and AP 2 changed with time. The ping-pong effect is quite obvious: neither AP 1 nor AP 2 gains a static throughput.

We then enabled the use of DC values. Fig. 7 shows the results. Clearly, ping-pong effects disappear and both APs share the same load.

V. Conclusions

We have presented an SNMP-based approach to congestion relief in IEEE 802.11 networks. Our approach applies to off-the-shelf APs and benefits already-deployed APs, as long as these AP support SNMP MIB-II. We have implemented our design and conducted experiments to measure its performance. Experimental results have clearly shown that our approach can distribute traffic load among available APs and thus significantly increase network throughput. The proposed scheme therefore resolves the unbalanced load problem that may arise in conventional IEEE 802.11 networks.

Acknowledgement

This work has been jointly supported by the National Science Council, Taiwan, under contract NSC-94-2213-E-216-001 and by Chung Hua University under grant CHU-94-TR-02.

References

[1] I. Papanikos and M. Logothetis, "A study on dynamic load balance for IEEE 802.11b wireless LAN," in *Proc. 8th Int'l Conf. on Advances in Communication and Control: Telecommunications/Signal Processing*, Crete, Greece, June 2001.

[2] Y. Fukuda, T. Abe, and Y. Oie, "Decentralized access point selection architecture for wireless LANs," in *Wireless Telecommunications Symposium*, May 2004, pp. 137–145.

[3] Y. Fukuda and Y. Oie, "Decentralized access point selection architecture for wireless LANs: Deployability and robustness," in *Proc. VTC2004-Fall*, Sept. 2004, pp. 1103–1107.

[4] H. Velayos, V. Aleo, and G. Karlsson, "Load balancing in overlapping wireless LAN cells," in *IEEE Int'l Conf. on Communications*, Sept. 2004, pp. 3833–3836.

[5] A. Balachandran, P. Bahl, and G. M. Voelker, "Hot-spot congestion relief in public-area wireless networks," in *Proc. 4th IEEE Workshop on Mobile Computing Systems and Applications*, 2002.

[6] K. McCloghrie and M. T. Rose, "Management Information Base for network management of TCP/IP-based internets: MIB-II," RFC 1213, Mar. 1991.

[7] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, ANSI/IEEE Std. 802.11, 1999.

[8] "Performance management: Best practices white paper," Cisco, Dec. 2003. [Online]. Available: http://www.cisco.com/warp/public/126/perfmgmt.pdf