

A Proposal of a Smart Access Point Allocation Algorithm for Scalable Wireless Mesh Networks

Kanako Uemura, Nobuo Funabiki, Toru Nakanishi *

Abstract—As a flexible, inexpensive, and extensive Internet access network, we have studied *WIMNET (Wireless Internet-access Mesh NETWORK)* composed of multiple access points (APs) as wireless mesh routers. WIMNET utilizes two types of APs to achieve the scalability with sufficient bandwidth while reducing costs. One is the expensive, programmable *smart AP (SAP)* that can use plural channels for wireless communications and has various functions for the Internet access. Another is the inexpensive, non-programmable *conventional AP (CAP)* that can use only one channel. To enhance the performance, the proper SAP allocation is critical, because the number of SAPs is much smaller than that of CAPs. In this paper, we present the definition of the SAP allocation problem, and its algorithm of extracting the optimum solution in terms of the newly defined cost function after finding the communication route configuration by applying our previous algorithms. We verify the effectiveness through simulations in three network topologies using the WIMNET simulator.

Keywords: *Wireless mesh network, smart access point, allocation, algorithm, communication route*

1 Introduction

Recently, *IEEE.802.11 wireless local area networks (WLANs)* have been extensively used as ubiquitous Internet access networks. The WLAN does not require the wired cable connecting a host with an access point (AP) that is a connection hub with a wired network. Thus, the WLAN has several advantages such as the low network installation cost, the easy host relocation, and the flexible communication area. As a result, the WLAN has been installed in a variety of places and institutions including governments, companies, homes, and schools. The public WLAN service has become popular in airports, stations, hotels, and even streets.

In the IEEE.802.11 WLAN, each AP can cover the limited area within approximately 100m distance from it due to the weak transmission signal. Thus, a single AP cannot offer the network service to the broad area, where multiple APs are necessary for the service. Convention-

ally, these APs are connected through wired cables. However, the cabling cost may impair the cost advantage of the WLAN, and even the cable may not be able to be laid down in places such as outdoors and old buildings. As a way to solve this problem, the APs are allocated in mesh, and the adjacent APs are communicated with wireless links in addition to wireless communications between APs and hosts. Communications between distance APs are realized by the multihop fashion, where their intermediate APs act as repeaters to reply packets. This multihop WLAN is called the *wireless mesh network* [1].

Currently, a variety of styles have been studied for the wireless mesh network. Among them, this paper adopts the one that uses only APs as wireless mesh routers and connects APs mainly on the MAC layer using the *wireless distribution system (WDS)*. At least one AP acts as a *gateway (GW)* to the Internet, where any host can connect to the Internet through a GW. For convenience, this wireless mesh network has been called *WIMNET (Wireless Internet-access Mesh NETWORK)* in our studies [2].

When the size of WIMNET is expanded with the increasing number of APs and hosts, it may meet two serious problems. One problem is the *increase of communication delay* due to the bandwidth shortage at the wireless links around the GW, because increasing traffics between the Internet and WIMNET must pass through the GW, whereas the bandwidth of those links is limited. Another problem is the *degradation of dependability and communication quality* due to the increasing interference between wireless links. As a result, the number of APs has been limited for WDS to avoid the unacceptable interference.

In order to solve these problems, we have proposed the hierarchical structure for the large-size WIMNET, which is composed of the *GW clusters* and the *WDS cluster*. As shown in Fig. 1, one GW cluster consists of the APs sharing the same GW, and one WDS cluster consists of the APs exchanging the same routing information for WDS. Besides, we have proposed the use of two types of APs, namely the expensive, programmable *smart AP (SAP)* and the inexpensive, non-programmable *conventional AP (CAP)*. The SAP can use plural channels for wireless communications by adding network interface cards (NICs), and has various functions for the Internet access [3]. On the other hand, the CAP can use only

*Dept. of Communication Network Engineering, Okayama University, 3-1-1 Tsushimanaka, Okayama, 700-8530, Japan Email: funabiki@cne.okayama-u.ac.jp

one channel. Then, one WDS cluster consists of one SAP as the *cluster head* and multiple CAPs where its number is limited, and one GW cluster consists of multiple WDS clusters. The WDS clusters are connected through SAPs with plural channels because their traffics are usually larger than those inside the WDS cluster. Then, the proper SAP allocation among APs becomes very important because it determines the performance and the cost of WIMNET.

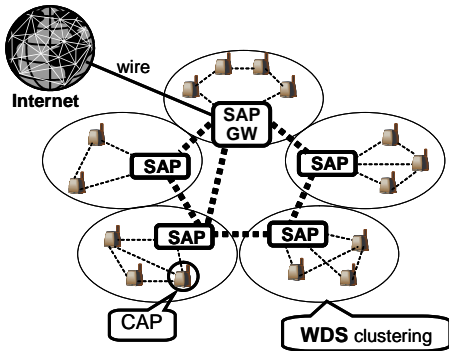


Figure 1: A GW cluster in WIMNET

In this paper, we define the SAP allocation problem for WIMNET, and present its algorithm. To maximize the performance of WIMNET, the minimization of the maximum transmission time or delay per channel is significant. The transmission time can be obtained by the traffics among the interfered links using the same channel. Because the interfered links can be obtained only after the routing and NIC/channel assignments are fixed, the proposed algorithm applies our previous routing tree and NIC/channel assignment algorithms in [4] for any feasible SAP allocation satisfying the constraints of the problem. Besides, the *maximum estimated SAP load* is adopted to prune undesirable SAP allocations offering unbalanced SAP loads [5] to reduce the computation time.

To evaluate our proposal, we simulate the throughput of the SAP allocation found by our algorithm, using the WIMNET simulator. As simulated network topologies, we adopt two artificial ones with small numbers of SAPs where APs are located on grids and the center AP is selected as the GW, and another practical one that models the JR Okayama station [6]. In addition, two different traffic patterns are prepared to each grid topology. Any simulation result indicates that the SAP allocation by our algorithm gives the maximum throughput that is similar to the all-SAP case. We conclude that using our proposal, a large scale WIMNET can be made up with a small number of expensive high-performance SAPs.

The rest of this paper is organized as follows: Section 2 defines the SAP allocation problem. Section 3 presents the algorithm. Section 4 shows evaluation results by simulations. Section 5 concludes the paper with some future

works.

2 Definition of SAP Allocation Problem

For the inputs to the SAP allocation problem, we assume that the AP network topology, the GW, the maximum expected number of associated hosts with each AP as loads, the interference among links, the transmission speed for each link, the WDS cluster size limit, and the channel interference matrix are given. As the output, the SAP allocation with the routing tree and NIC/channel assignments is requested to minimize the maximum transmission time. Then, the SAP allocation problem can be formulated as follows:

A. Input:

- the AP network topology: $G = (V, E)$
 - the number of APs: $N = |V|$
 - the links between APs: E
 - the interference among links: $D = [d_{ijpq}]$, where $d_{ijpq} = 1$ if two links, $AP_i \rightarrow AP_j$ and $AP_p \rightarrow AP_q$, are interfered, and $d_{ijpq} = 0$ otherwise
 - the bandwidth of $link_{ij}$: s_{ij} (Mbps)
 - the maximum expected number of associated hosts with AP_i : h_i
 - the Internet gateway: $g (\in V)$
- the number of SAPs: M
- the WDS cluster size (the maximum number of CAPs in one WDS cluster): $WSIZE$
- the channel interference matrix: $C = [c(i, j)]$
- the number of channels: P
- the maximum number of NICs per SAP: Q

B. Output: the SAP allocation with the routing tree and NIC/channel assignments

C. Constraints: The following three conditions must be satisfied in the feasible solution:

- (1) The gateway must be SAP.
- (2) No CAP must exist along the routing path between the GW and any SAP.
- (3) Any CAP must have at least one connectable SAP as the cluster head.

The *connectable SAP* represents a SAP that exists along the shortest path between the CAP and the GW, or exists within four hops from the CAP. The latter condition is necessary to increase the number of cluster head candidates for CAPs to obtain feasible solutions.

D. Objective: The cost function E_{traf} representing the maximum transmission time should be minimized:

$$E_{traf} = \max_{(i,j)} \left[t_{ij} + t_{ji} + \sum_{p=1}^N \sum_{q=1}^N t_{pq} c(y_{ij}, y_{pq}) \right] \quad (1)$$

$d_{ijpq} = 1 \vee d_{jipq} = 1$

where t_{ij} represents the traffic along the link from $AP_i \rightarrow AP_j$, and y_{ij} represents its assigned channel.

3 Proposal of SAP Allocation Algorithm

3.1 SAP Allocation Algorithm

Previously, we have proposed the routing tree algorithm and the NIC/channel assignment algorithm for WIMNET [4]. The former algorithm finds the communication route between the GW and the APs for a given AP network with the fixed SAPs. The latter algorithm finds the number of NICs assigned to each SAP and the channel assignment to each NIC for a given routing tree. Then, because the cost function E_{traf} in the SAP allocation problem can be calculated only after the routing tree and the NIC/channel assignment are found, our algorithm applies these algorithms to any feasible SAP allocation for a given AP network, and selects the best one that minimizes E_{traf} . Here, to reduce the computation time, our algorithm discards undesirable SAP allocations by adopting the *maximum estimated SAP load* before applying previous algorithms. The following procedure describes the details of the SAP allocation algorithm:

- (1) Calculate the lower bound N_{LB}^{SAP} on the number of SAPs, which is equal to the number of WDS clusters. If the input number of SAPs M is smaller, then set $M = N_{LB}^{SAP} = \lceil \frac{N}{WSTZE} \rceil$.
- (2) Generate a new SAP allocation by selecting M APs for SAPs among N APs.
- (3) Check the feasibility of the SAP allocation of N_{LB}^{SAP} of satisfying the three constraints. If it is not feasible, go to (8).
- (4) Calculate the *maximum estimated SAP load* in Sect. 3.2. If it is larger than the threshold there, go to (8).
- (5) Apply the routing tree and NIC/channel assignment algorithms in [4]. If no feasible solution is obtained, go to (8).
- (6) Calculate the cost function E_{traf} .

- (7) Update the best-found solution if E_{traf} in (6) is smaller than the best one.
- (8) Terminate the procedure if every SAP allocation is generated in (2). Otherwise, go to (2).

3.2 Maximum Estimated SAP Load

As the number of APs increases for a large-size WIMNET, the number of SAP allocations increases exponentially. Then, the computation time becomes unacceptably long, where the routing tree and NIC/channel assignment algorithms inhibitory requires the long time. To avoid this situation, the *maximum estimated SAP load* is calculated for each SAP allocation before their applications, and if the value is larger than the threshold, it is discarded because traffics are concentrated into a specific SAP that may become the bottleneck of WIMNET.

3.2.1 SAP Selection Weight by Hop Count

As the WDS cluster head of a CAP, a SAP with the smaller hop count (number of hops) has the larger possibility than a SAP with the larger hop count due to the delay. Thus, the *SAP selection weight* by hop count is calculated for each pair of a SAP and a CAP to represent the possibility of being the same cluster by $W_k = 1/2^k$.

3.2.2 Procedure for Maximum Estimated SAP Load

For each SAP in a SAP allocation, the *maximum estimated SAP load* is calculated by the following procedure:

- (1) Find the connectable SAPs for each CAP in the SAP allocation.
- (2) Calculate the weighted average of the *SAP selection weights* among the connectable SAPs for each CAP.
- (3) Multiply the maximum expected number of associated hosts h_i to this weighted average for each SAP.
- (4) For each SAP, sum up the values of W_{kin} of all the CAPs that can select this SAP as the cluster head, which becomes the *estimated SAP load*.
- (5) Select the maximum value of (4) among all the SAPs in the SAP allocation for the *maximum estimated SAP load*.

3.2.3 Example of Estimated SAP Load

Figure 2 illustrates an example of calculating the *estimated SAP load* for CAP-A. The three SAPs {1, 2, 3}

are connectable for CAP-A. Because the hop count from CAP-A is three, two, and three to each SAP, the corresponding *SAP selection weights* are given by: $1/2^3$, $1/2^2$, $1/2^3$. Because h_i for CAP-A is 10, the estimated load for each of the three SAPs by CAP-A is 2.5, 5, and 2.5 as shown in Fig. 2. Then, after calculating them for every CAP, the *estimated SAP load* is calculated by summing up them for each SAP.

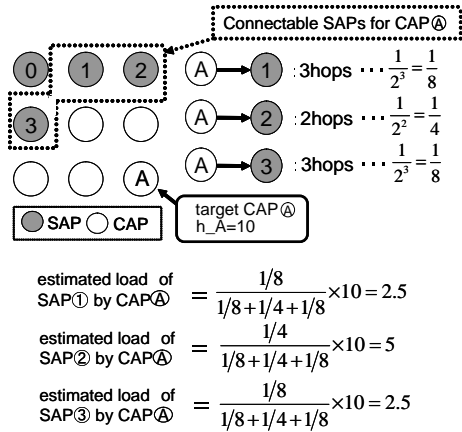


Figure 2: Example of estimated SAP load calculation.

3.2.4 Threshold for Traffic Concentration

The *maximum estimated SAP load* is compared with the given threshold Th to judge whether the SAP allocation may cause the traffic concentration into a specific SAP or not. If it exceeds the threshold, the SAP allocation is discarded. For this threshold, the twice of the average SAP load is used:

$$Th = \left(\frac{\sum_{i=1}^N h_i}{M} \right) \times 2. \quad (2)$$

4 Evaluation by Simulations

In this section, we evaluate the proposed SAP allocation algorithm in terms of the computation time and the throughput by packet transmission simulations using the WIMNET simulator. For simulations, we prepare three AP network topologies with different communication loads. Each simulation assumes that at the beginning, every host poses 125 packets to the GW (Internet) and the GW does 1,000 packets to every host. The bandwidth is $30Mbps$ for any link between two APs and $20Mbps$ for that between an AP and a host. When multiple links within interference ranges try to be activated simultaneously, randomly selected one link among them is successfully activated, and the others are inserted into

waiting queues. After every packet reached the destination, the throughput is calculated by dividing the total packet size with the simulation time.

4.1 Simulated Instances

topology 1 adopts the grid one in Fig. 3 that has been often used in mesh network studies. $25 (= 5 \times 5)$ APs are allocated on the grid, where the center AP is selected as the GW. Each AP can communicate wirelessly with its four neighbor APs. As network loads, two different traffic patterns are prepared, where the black circle represents a crowded AP associating 8 hosts for *pattern 1* and 10 hosts for *pattern 2*, and the white circle represents an AP associating only 1 host, so that about 100 hosts totally exist in each pattern. The WDS cluster size $WSIZE$ is set 8, which indicates that the lower bound on the number of clusters N_{LB}^{SAP} is 4 from $\lceil \frac{25}{8} \rceil = 4$. Actually, we select $m = 5$ because it gave the similar throughput to the case where every AP is selected as a SAP in our preliminary experiments.

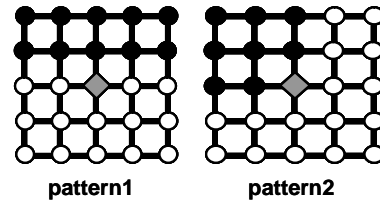


Figure 3: Network topology and traffic patterns for *topology 1*.

topology 2 adopts the same grid topology as *topology 1* with increasing 49 APs, to investigate the scalability of our algorithm. The number of crowded APs (black circles) is also increased, and the number of SAPs is doubled.

topology 3 in Fig. 4 adopts a topology that models the second floor at JR Okayama station, to consider the more practical field. The AP allocation and network loads are given by considering the floor structure and the flow of people there. We note that the number inside a circle represents the maximum expected number of hosts at the AP. The AP around the field center is selected as the GW where no host is associated.

4.2 Evaluation of Cost Function

First, we evaluate the validity of the cost function E_{traf} in terms of the throughput obtained by the WIMNET simulator. For each simulation, we use the solution (the SAP allocation, the routing tree and NIC/channel assignments) of our algorithm as the WIMNET configuration. Here, the maximum number of NICs per SAP Q is four for *topology 1* and *topology 2* and five for *topology 3*, which indicates that the largest possible throughput

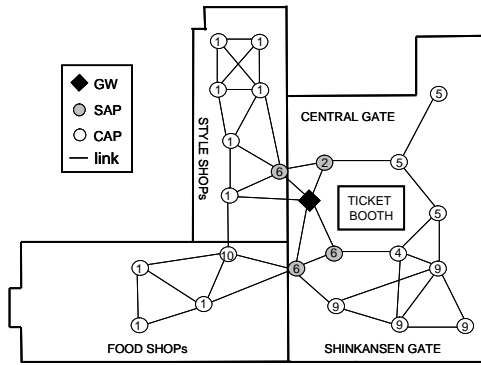


Figure 4: Network topology and traffic patterns for *topology 3*.

is 120Mbps for the first two topologies and 150Mbps for the last. Figures 5 and 6 show the relationship graph between the cost function and the throughput at each traffic pattern for *topology 1*, Figs. 7 and 8 do it for *topology 2*, and Fig. 9 does it for *topology 3*, respectively. In any graph, the throughput becomes maximum when the cost function is minimum, which supports the validity of our cost function.

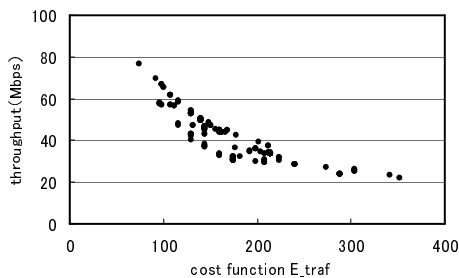


Figure 5: Relationship between cost function and throughput at *pattern 1* for *topology 1*.

Figure 10 illustrates the SAP allocation and the routing tree at each traffic pattern for *topology 1* by our algorithm. Any solution satisfies the constraints of the SAP allocation problem and balances the loads well between SAPs.

4.3 Evaluation of Maximum Estimated SAP Load

Then, we evaluate the effectiveness of the use of the *maximum estimated SAP load* in reducing the computation time of our algorithm.

First, we verify the validity of the *SAP selection weight by hop count*. Table 1 shows the relationship between the hop count and the selected rate such that a certain SAP/CAP pair is actually included in the routing tree by

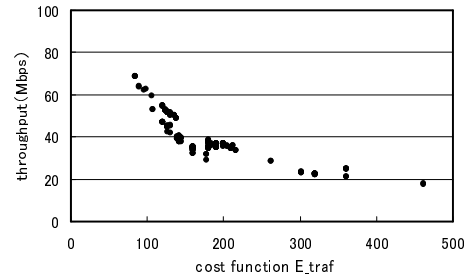


Figure 6: Relationship between cost function and throughput at *pattern 2* for *topology 1*.

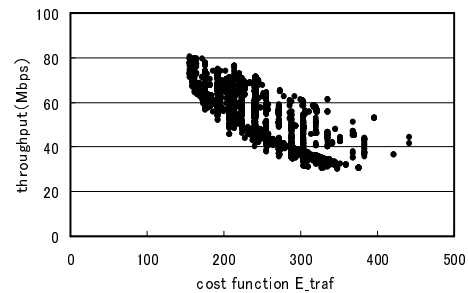


Figure 7: Relationship between cost function and throughput at *pattern 1* for *topology 2*.

our algorithm. This table indicates that the selected rate roughly becomes 1/2 every time the hop count increases by one, which supports the validity of our *SAP selection weight by hop count*.

Then, to evaluate the effectiveness of the *maximum estimated SAP load* in terms of the computation time reduction, we count the number of SAP allocations where the routing tree and NIC/channel assignment algorithms are applied, when each of the three constrains ((1), (2), (3)) and the maximum estimated SAP load ((4)) is sequentially applied. Table 2 shows that for *topology 1* and *topology 3* with a fewer numbers of APs and SAPs, the *maximum estimated SAP load* reduces it into about 2/5 of the result by the constraints, and for *topology 3*, it does in about 1/6. Thus, the use of the *maximum estimated SAP*

Table 1: Relationship between hop count and SAP/CAP pair selected rate.

	1 hop	2 hops	3 hops	4 hops
<i>topol. 1, pat. 1</i>	80.0%	38.4%	7.1%	0.0%
<i>topol. 1, pat. 2</i>	72.2%	33.3%	22.2%	8.3%
<i>topol. 2, pat. 1</i>	58.8%	30.7%	21.2%	8.3%
<i>topol. 2, pat. 2</i>	64.7%	31.5%	20.4%	9.6%
<i>topol. 3</i>	80.0%	36.8%	17.6%	0.0%

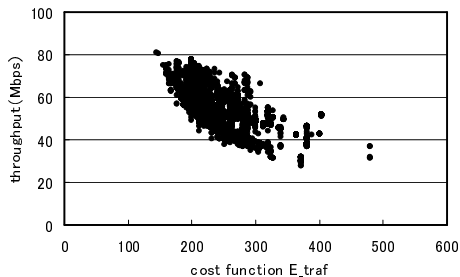


Figure 8: Relationship between cost function and throughput at *pattern 2* for *topology 2*.

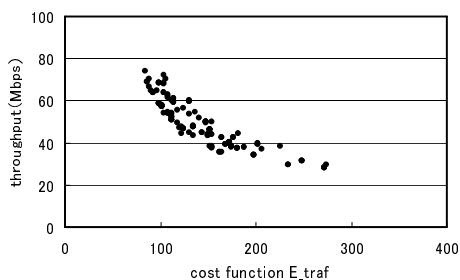


Figure 9: Relationship between cost function and throughput for *topology 3*.

load is very effective to reduce the computation time.

Table 3 shows the total computation time of our algorithm on a PC with *Core 2 Duo 2.67 GHz* for CPU, *3.25 GB* for main memory, and *Linux 2.6.16* for OS. With the help of the *maximum estimated SAP load*, our algorithm can run within acceptable time for a large-scale WIMNET.

5 Conclusion

This paper presented the definition of the smart access point (SAP) allocation problem for the scalable wireless Internet-access mesh network *WIMNET* and its al-

Table 2: Number of examined SAP allocations by algorithm.

	(1)	(1),(2)	(1)-(3)	(1)-(4)
<i>topol. 1, pat. 1</i>	251	243	243	147
<i>topol. 1, pat. 2</i>	251	243	243	107
<i>topol. 2, pat. 1</i>	173252	108544	33208	5384
<i>topol. 2, pat. 2</i>	173252	108544	33208	6513
<i>topol. 3</i>	229	203	203	164

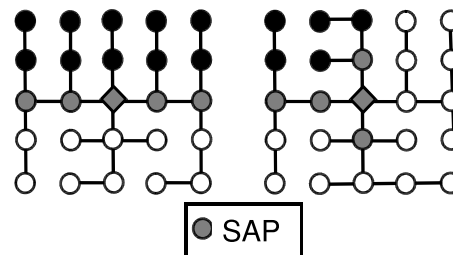


Figure 10: SAP allocation and routing tree for *topology 1*.

Table 3: Algorithm computation time (seconds).

<i>topol. 1 pat. 1</i>	<i>topol. 1 pat. 2</i>	<i>topol. 2 pat. 1</i>	<i>topol. 2 pat. 2</i>	<i>topol. 3</i>
22	18	4692	4941	20

gorithm adopting the *maximum estimated SAP load* for the computation time reduction. The effectiveness of our proposal was verified through simulations in three network topologies using the WIMNET simulator. Our future works may include extensions of the SAP allocation algorithm to dynamic changes of network loads and network configurations by failures of links or access points, and evaluations in the real network after the SAP implementation.

References

- [1] Akyildiz, I. F., Wang, X., Wang, W., "Wireless mesh networks: a survey," *Comput. Network*, vol. 47, no. 4, pp. 445-487, March 2005.
- [2] Farag, T., Funabiki, N., Nakanishi, T., "An access point allocation algorithm for indoor environments in wireless mesh networks", *IEICE Trans. Commun.*, vol. E92-B, no. 3, pp. 784-739, March 2009.
- [3] Hirakata, K., Horiuchi, T., Funabiki, N., Nakanishi, T., "A construction of the smart access point for practical wireless mesh networks," *IEICE Tech. Report*, vol. NS2008-99, pp. 63-68, Nov. 2008.
- [4] Uemura, K., Funabiki, N., Nakanishi, T., "A communication route optimization algorithm for scalable wireless mesh networks," *IEICE Trans. B*, vol. J92-B, no. 9, Sep. 2009.
- [5] Hsiao, P.-H., Hwang, A., Kung, H. T., Vlah, D., "Load-balancing routing for wireless access networks," *Proc. IEEE INFOCOM*, vol. 2, pp. 986-995, April 2001.
- [6] JR Okayama Station, <http://www.jr-odekake.net/eki/premises.php?id=0650608>.