

Article

A Study of the Active Access-Point Configuration Algorithm under Channel Bonding to Dual IEEE 802.11n and 11ac Interfaces in an Elastic WLAN System for IoT Applications

Sujan Chandra Roy ¹, Nobuo Funabiki ^{1,*}, Md. Mahbubur Rahman ², Bin Wu ¹, Minoru Kuribayashi ¹ and Wen-Chung Kao ³

¹ Graduate School of Natural Science and Technology, Okayama University, Okayama 700-8530, Japan; p7ku89oh@s.okayama-u.ac.jp (S.C.R.)

² Department Electrical and Electronic Engineering, Jatiya Kabi Kazi Nazrul Islam University, Trishal, Mymensingh 2224, Bangladesh

³ Department of Electrical Engineering, National Taiwan Normal University, Taipei 106, Taiwan

* Correspondence: funabiki@okayama-u.ac.jp

Abstract: Currently, *Internet of Things (IoT)* has become common in various applications, including smart factories, smart cities, and smart homes. In them, *wireless local-area networks (WLANs)* are widely used due to their high-speed data transfer, flexible coverage ranges, and low costs. To enhance the performance, the WLAN configuration should be optimized in dense WLAN environments where multiple *access points (APs)* and hosts exist. Previously, we have studied the *active AP configuration algorithm for dual interfaces* using *IEEE802.11n* and *11ac* protocols at each AP under *non-channel bonding (non-CB)*. In this paper, we study the algorithm considering the *channel bonding (CB)* to enhance its capacity by bonding two channels together. To improve the throughput estimation accuracy of the algorithm, the *reduction factor* is introduced at contending hosts for the same AP. For evaluations, we conducted extensive experiments using the *WIMENT simulator* and the *testbed system* using *Raspberry Pi 4B APs*. The results show that the estimated throughput is well matched with the measured one, and the proposal achieves the higher throughput with a smaller number of active APs than the previous configurations.

Keywords: *Internet of Things*; WLAN; access-points configuration; dual interface; *channel bonding*; WIMNET; Raspberry Pi 4B; IEEE802.11n; 11ac



Citation: Roy, S.C.; Funabiki, N.; Rahman, M.M.; Wu, B.; Kuribayashi, M.; Kao, W.-C. A Study of the Active Access-Point Configuration Algorithm under Channel Bonding to Dual IEEE 802.11n and 11ac Interfaces in an Elastic WLAN System for IoT Applications. *Signals* **2023**, *4*, 274–296. <https://doi.org/10.3390/signals4020015>

Academic Editors: Yiming Huo and Minh Tu Hoang

Received: 30 November 2022

Revised: 28 February 2023

Accepted: 20 March 2023

Published: 3 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Nowadays, *Internet of Things (IoT)* is gaining popularity all over the world with increasing the number of devices connected to the Internet. IoT has been widely deployed in various applications, including smart factories, smart cities, smart homes, smart grid systems, smart farms, smart warehouses, and smart healthcare systems [1–6].

In IoT application systems, device-to-device connections are usually made through wireless networks, such as Wireless Sensor Networks (WSNs), Wireless Local Area Networks (WLANs), Long Term Evolution (LTE), and Global System for Mobile Communication (GSM) [7–11]. Among them, many IoT application systems adopt IEEE 802.11 WLANs, as shown in Figure 1 [8,12–14].

Currently, a huge number of IoT devices are connected to the Internet through IEEE 802.11 WLANs, because their popularity has remarkably increased around the world due to high-speed data transfer, flexible coverage ranges, low costs, and easy installation. In WLAN, hosts are connected to the *access point (AP)* wirelessly, which makes it extensible and flexible compared with wired LAN. As a result, WLAN services are available in offices, schools, and public places, such as hotels, airports, malls, stations, and even trains and airplanes [15–17]. In traditional WLANs, hosts communicate with servers over the Internet,

while IoT devices communicate with servers. In this paper, a *host* represents a smartphone, a PC, a tablet, or an IoT device that will be accessed to the Internet through WLAN.

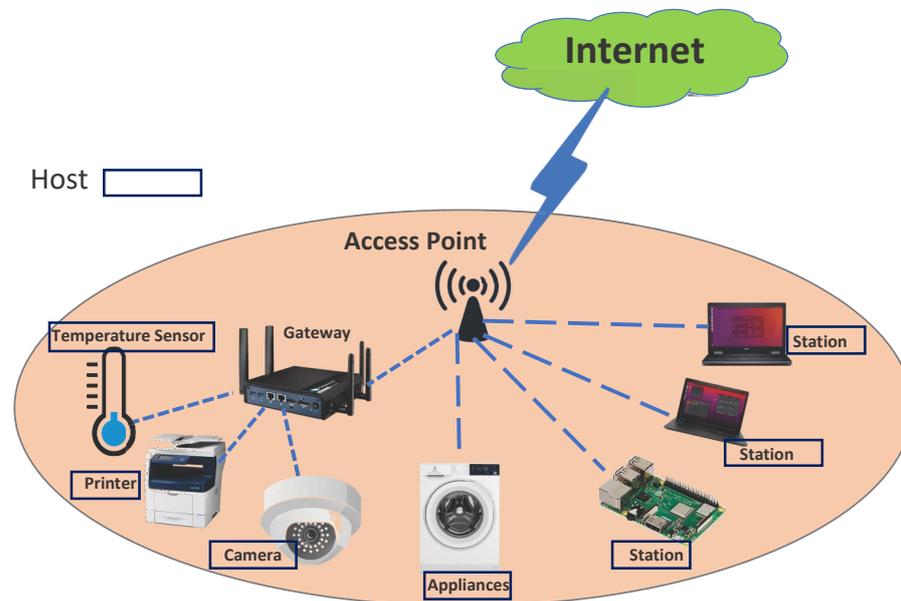


Figure 1. WLAN for an IoT system.

In a network field, multiple APs are often deployed to support a large number of users. Such situations are called *dense WLAN environments*. In dense WLAN environments, network performance can be degraded by radio-signal interference between nearby APs and hosts. To avoid the performance degradation as best as possible, the network configuration of the WLAN, such as activations/deactivations of APs and associations of hosts with active APs, should be properly optimized, according to traffic demands. If redundant APs exist, they should be deactivated to reduce energy consumptions [18–21]. Therefore, we have proposed the active AP configuration algorithm to dynamically optimize the network configuration by activating or deactivating APs according to traffic demands [22].

The throughput performance in a WLAN is affected by various factors in an indoor environment. Therefore, we have studied the *throughput estimation model* with the *parameter optimization algorithm* [23], and the *AP setup optimization approach* in terms of the orientation and the height of the device [24].

The number of users using WLANs to access the Internet is increasing, resulting in fast growth of traffic. In order to meet the ever-increasing traffic, *channel bonding (CB)* was introduced in the *IEEE 802.11n* amendment for improving the network performance of WLANs. In the CB, two adjacent 20 MHz channels are bonded together to form one 40 MHz channel [16,25,26]. The *IEEE 802.11ac* and the *IEEE 802.11ax* may further extend the channel-bonding feature by allowing 80 MHz or 160 MHz channels with four or eight single channels, respectively, refs. [27–29]. However, we did not consider them, since they are still not common in WLAN. In addition, our experiments using IEEE 802.11ax devices [30] show that the throughput performance does not reach the expected level. Therefore, their use will be in future works.

In [31], using *orthogonal frequency division multiplexing (OFDM)*, the CB raises the number of sub-carriers from 52 to 108. To get higher throughput in this wide-band CB, the OFDM can transmit data using a multiple of narrow-band carriers.

However, the CB reduces the number of non-interfered channels and may increase interference between nearby APs and hosts due to wider channel bands in dense WLANs. In [32], Barrachina-Muñoz et al. demonstrated that a CB link can decrease the *signal-to-interference-plus-noise ratio (SINR)* at a distant host from the AP due to the wider channel bandwidth. A lower SINR may result in a slower *modulation and coding scheme (MCS)* that

will cause lower throughput. The wider bandwidth of the CB link may also cause the *hidden terminal problem*.

A dual interface at an AP using 11n and 11ac at 2.4 GHz and 5 GHz frequency bands is an efficient approach to enhance the network's performance while avoiding interference. It can reduce the number of active APs, which can further reduce the interference. Therefore, we have studied the *active AP configuration algorithm* considering the *dual interface* [33] of a *Raspberry Pi* [34] AP adopting an external USB interface Archer T4U [35]. In this dual interface, the embedded one is used for 11n, and the external one is for 11ac.

However, the previous algorithm limits the *non-channel bonding (non-CB)*. Thus, it should be extended to consider the CB. In addition, the algorithm selects the active APs and their host associations by estimating the throughput of a host using the *throughput estimation model*, where the estimation accuracy of the model is critical to determine the network performance. Thus, it should be improved under higher interference by the CB.

In this paper, we study the *active AP configuration algorithm* for the *dual interface* with *channel bonding (CB)* at the same AP. The goal of this algorithm is to find the AP configuration that minimizes the number of active APs in the WLAN under the minimum host throughput constraint for every host. The *dual interfaces* at the AP will operate at different frequency bands, which allows the AP to communicate with two hosts simultaneously without any interference between them. As a result, the *dual interface* can reduce interference between neighboring APs in the network and can solve the abovementioned problems in the CB. The main contributions of this paper are summarized as follows:

1. We investigate the protocol selection at each Wi-Fi interface in a WLAN to make the best use of them in the proposed access point (AP) with the *dual interfaces* in the testbed system. From experiments, we show that the selection of 11n at 2.4 GHz at the embedded interface of *Raspberry Pi* and of 11ac at 5 GHz for the external USB interface *Archer T4U* has better performances.
2. We present the modification of the *active AP configuration algorithm* to take into account the use of the *dual interfaces* and the *channel bonding (CB)*, where the interfaces operate at different frequency bands with IEEE 802.11n and 11ac protocols, and the AP can communicate with two hosts simultaneously without any interference between them.
3. We incorporate the *throughput reduction factor* into the *throughput estimation model* to improve the estimation accuracy when multiple hosts are associated with the same AP.
4. We evaluated the proposal through both simulations using the *WIMENT simulator* [36] and experiments using the testbed system in real fields. The testbed system involved a *Raspberry Pi* with an external *Archer T4U* for the AP and *Linux PCs* for the hosts.

The rest of this paper is organized as follows: Section 2 presents related works in the literature. Section 3 reviews our previous works. Section 4 discusses the interface selection through experiments. Section 5 presents modifications of the active AP configuration algorithm. Section 6 evaluates the proposal through simulations and experiments. Finally, Section 7 provides concluding remarks with future works.

2. Related Works

In this section, we briefly discuss related works in the literature.

In [37], Yan et al. presented an algorithm for AP deployment based on physical distance and channel isolation (DPDCI) in urban environments, which can effectively reduce interference and improve the system's overall throughput. The algorithm optimizes the placement of APs and allocates the channel to APs by adopting a genetic algorithm.

In [38], Tang et al. proposed a new optimal AP deployment (OAD) algorithm based on fuzzy C-clustering, which provides good performance in the maximum number of WLAN users and balanced the traffic load among the APs.

In [39], Tian et al. proposed a genetic algorithm Cramer–Rao lower bound (GA-CRLB) method for finding optimal AP positioning that maximizes localization accuracy and coverage simultaneously. This method adopted CRLB as an evaluation methodology for

evaluating the localization accuracy of the deployment plan, and then the genetic algorithm was used to rapidly find the optimal AP deployment strategy.

In [40], Wan et al. jointly presented a user association and bandwidth allocation algorithm called the bandwidth-constrained association (BCA) to maximize the total network throughput while satisfying user device bandwidth requirements. They adopted an efficient polynomial time method for producing the best user association.

In [41], Islam et al. presented a joint AP association and bandwidth-allocation scheme in order to optimize network throughput and utilization taking into account access points' (APs) capacity and user demand. They adopted the rounding and bandwidth allocation algorithm (RBAA) for this model.

In [42,43], Zhi et al. and Liu et al. proposed novel AP deployment methods by using a genetic algorithm (GA) and fruit fly optimization algorithm (FOA), respectively. In both proposals, the AP locations and the transmit powers of all the activated APs are optimized while ensuring complete coverage of the desired service area.

In [44], Liu et al. proposed analytical expressions to measure the throughput performance of WLAN while considering the inter-network interference. They adopted the airtime concept to express the simultaneous transmission time and the carrier-sensing duration to understand the carrier-sensing duration for this proposal.

In [45], Kong et al. proposed a method for optimizing AP deployment based on the multi-purpose particle swarm optimisation algorithm (MOPSO). First, the performance of a single AP was evaluated using the random geometry theory to estimate the number of APs that should be deployed in the WLAN, based on the user's service needs. Then, the proposed scheme uses the MOPSO algorithm to obtain the optimal position and optimal transmit power of the APs. Finally, a greedy algorithm is used in order to remove the redundant APs.

In [46], Qiu et al. presented a joint optimization method for the placement of access points, power assignment, channel assignment, and resource unit assignment in dense IEEE 802.11ax WLAN. The objective of the proposal is to reduce the cost regarding the number of access points (APs) while meeting the requirements of specified failure tolerance and a specified two-tier flow. The authors designed a heuristic algorithm based on polynomial time complexity in order to find high-quality solutions and compared the throughput results to the greedy and random solution.

In [47], Abusubaih et al. analyzed the ability of IEEE 802.11n dual-band APs to support heterogeneous client adapters and applications. By testbed experiments, they concluded that dual-band APs can effectively reduce the negative effects of heterogeneity on clients' network adapters.

In [48], Nishat et al. proposed a QoS differentiation scheme called Slickfi for high-speed WLANs. It aims to enhance the real-time application performance while maintaining the high network throughput. It achieves the high performance by isolating real-time traffic and data traffic using two radios in APs managing adjacent-channel interference and allowing the per-frame channel width adaptation to maximize the network throughput.

Table 1 overviews the comparison between our proposal and the existing approaches in the literature. Most of them improve the network performance with a single interface at the AP, whereas none of them adopt the IEEE 802.11n and 11ac dual interfaces at the AP to minimize the number of active APs according to the user demand and to improve the network's performance.

Table 1. Comparison between our proposal and existing works.

Reference	Goal	Method	Satisfy USER Requirement	Dense Wi-Fi	Dual Interface	Minimize AP Cost	Evaluation
proposal	minimize the number of active APs	greedy algorithm and local search method	yes	yes	yes	yes	simulation and testbed experiment
[37]	optimizes the APs placement	genetic algorithm	no	yes	no	no	simulation
[38]	adjust the AP deployment	fuzzy C-clustering	yes	yes	no	yes	simulation
[39]	minimize the number of APs	genetic algorithm and cramer-rao lower bound method	yes	yes	no	yes	simulation
[40]	optimal userassociation and maximizes the total network throughput	polynomial-time algorithm	yes	yes	no	no	simulation
[41]	maximizes the network throughput and AP utilization	rounding and bandwidth allocation algorithm	yes	no	no	no	simulation
[42]	minimize the number of APs	genetic algorithm	no	yes	no	yes	simulation
[43]	minimize the number of APs	fruit fly optimization algorithm	no	yes	no	yes	simulation
[44]	throughput analysis take into account the effects of inter-network interference	airtime concept	no	yes	no	no	simulation
[45]	minimize the number of APs	multi-objective particle swarm optimization (MOPSO) algorithm and greedyalgorithm	yes		no	yes	simulation
[46]	minimize the number of APs	polynomial time heuristic algorithm	yes	yes	no	yes	simulation
[47]	investigate the ability of IEEE 802.11n dual-band APs	-	no	no	no	no	testbed experiment
[48]	maximize the performance of real-time application and network throughput	QoS differentiation scheme (Slickfi)	no	no	yes	yes	testbed experiment

3. Review of Previous Works

In this section, we review our previous works related to this paper.

3.1. Throughput Estimation Model

The throughput estimation model is used to estimate the throughput of a wireless communication link between a source node and a destination node in a WLAN. First, it estimates the *received signal strength (RSS)* using the log-distance path-loss model at the destination node [49]. Then, the sigmoid function is used to convert the RSS into throughput.

3.1.1. Received Signal Strength Estimation

First, the Euclidean distance d (m) is calculated for each link (AP/host pair) by:

$$d = \sqrt{(AP_x - H_x)^2 + (AP_y - H_y)^2} \tag{1}$$

where AP_x and H_x represent the x coordinate of the AP and the host; and AP_y and H_y represent the y coordinate, respectively. After that, the received signal strength (RSS) P_d (−dBm) is estimated at the host by:

$$P_d = P_1 - 10\alpha \log_{10}d - \sum_k n_k W_k \tag{2}$$

where P_1 represents *RSS* from the AP to the host at the 1 m distance when there is no obstacle; α represents the path loss exponent; n_k is the number of *type_k* obstacles or walls along the route between the AP and the host; and W_k represents the signal attenuation ratio (dBm) for the *type_k* ($k = 1, 2, \dots, 6$) obstacle. It is noted that in a building, there are several types of walls that have different thicknesses and materials, which make the different signal attenuation factors.

The value of α depends on the specific environment in which the radio signal is propagating. It represents the amount of loss that the signal will experience as it propagates through the environment. $type_k$ obstacles represent the obstacles or walls that have the k -th signal attenuation factor W_k . In this paper, we consider six types of obstacles, namely, the corridor wall for W_1 , the partition wall for W_2 , the intervening wall for W_3 , the glass wall for W_4 , the elevator wall for W_5 , and the door for W_6 [24].

3.1.2. Throughput Estimation from Received Signal Strength

From *received signal strength* P_d , the throughput/link speed is calculated by using the *sigmoid function* [24]:

$$tp = \frac{a}{1 + e^{-\left(\frac{120+P_d}{c}-b\right)}} \tag{3}$$

where tp represents the estimated throughput (Mbps); P_d does RSS (−dBm) at the host; and a , b , and c are constant parameters obtained using measurements. These values are optimized by the *parameter optimization tool* [23] in Table 2.

Table 2. Parameter optimization results under CB for 11n and 11ac.

Parameter	Field #1				Field #2	
	Group A		Group B		11n	11ac
	11n	11ac	11n	11ac		
P1	−28.9	−31.0	−31.1	−36.1	−28.5	−30.5
α	2.20	2.15	1.6	2.18	1.7	2.0
W1	7.21	2.1	6.5	4.2	6.5	2.3
W2	6.9	8.5	3.5	4.1	4.2	6.4
W3	3.4	3.7	3.5	4.4	3.1	1.8
W4	4.7	1.8	3.5	4.55	1.5	4.2
W5	2.11	7.0	2.5	2.1	2	4.3
W6	2.5	1.5	1.5	1.5	2	5.3
a	63.5	133	66	77	65	134.5
b	62	58	70	54.1	62	58.5
c	6.78	6.30	5.2	5.2	6.78	6.25

3.1.3. Throughput Reduction Factor

When multiple hosts are communicating with an AP simultaneously, the *throughput reduction factor* is introduced in our model to improve the estimation accuracy [50]. It is calculated by:

$$tp \times srf(m) \tag{4}$$

where m is the number of hosts communicating with AP, tp represents the estimated single link throughput between AP_i and $host_j$, and $srf(m)$ is the empirically derived contention factor at AP_i among the associated hosts for sending data through the CSMA/CA protocol. $srf(m)$ is calculated by

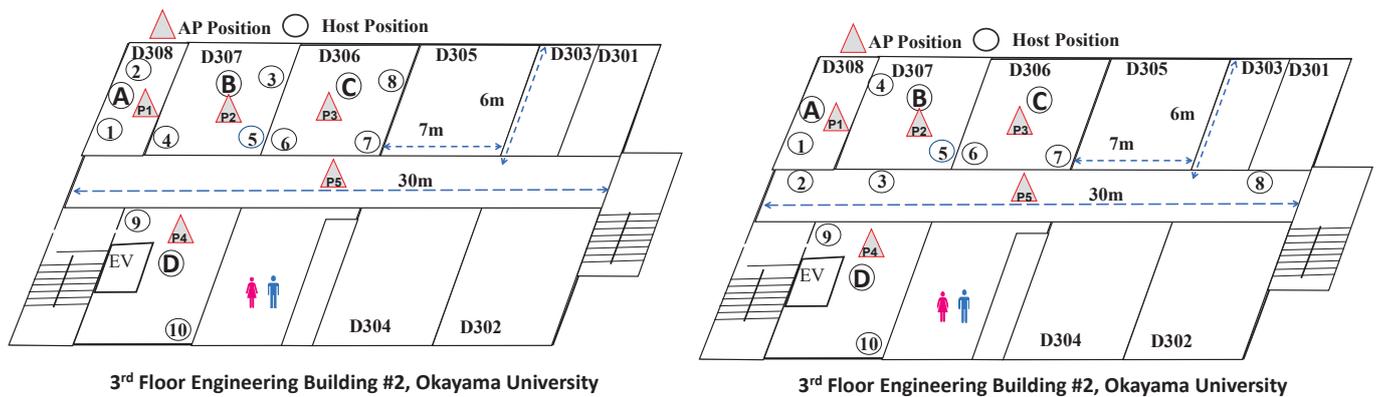
$$srf(m) = \left(\frac{1}{m + \frac{0.1(m-1)}{4}} \right) \times 1 - (0.1 \times m - 1) \tag{5}$$

3.1.4. Model Parameter Optimization

The throughput estimation model has multiple parameters whose value determines the accuracy of the estimation. The *parameter optimization tool* is adopted to optimize these values. It adopts a local search method that adopts the *tabu table* and the *hill-climbing function* together in order to avoid a local minimum convergence [23].

Table 2 demonstrates the summary of the parameter optimization results for throughput estimation models that are used for two fields in this paper for 11n and 11ac, *Field #1* and *Field #2* [24]. *Field #1* represents the *third floor of Engineering Building #2* in Figure 2.

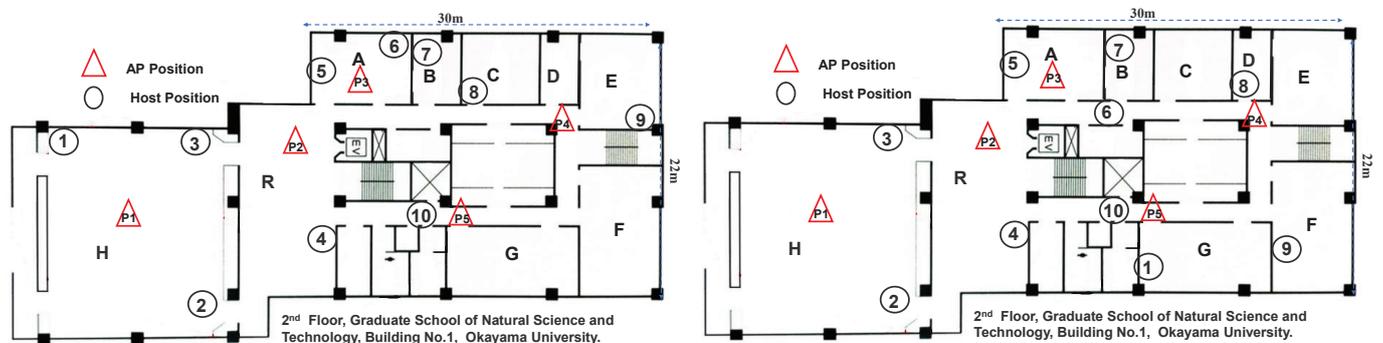
Field #2 represents the second floor of Graduate School Building at Okayama University, as shown in Figure 3.



(a) Topology 1 with 5 APs and 10 hosts.

(b) Topology 2 with 5 APs and 10 hosts.

Figure 2. Experimental field #1.



(a) Topology 3 with 5 APs and 10 hosts.

(b) Topology 4 with 5 APs and 10 hosts.

Figure 3. Experimental field #2.

3.2. Elastic WLAN System

The *elastic WLAN system* has been designed and implemented to dynamically optimize the network topology and the configuration in accordance with the network conditions. Figure 4 depicts an example topology of the system.

The elastic WLAN system implementation adopts the *management server* to manage and control the hosts and the APs by running the *active AP configuration algorithm* [22]. This server has not only administrative access to all devices in the network but also controls the overall network using the following three procedures:

1. It explores all network devices and also collects the information that are required for the active AP configuration algorithm.
2. After that, it executes the active AP configuration algorithm through the derived inputs from the previous step. The output of the algorithm contains the list of the active APs, the host associations, and the assigned channels of each active AP.
3. Finally, it applies the algorithm output into the network by changing the host associations, activating or deactivating the specified APs, and allocating the channels to the APs.

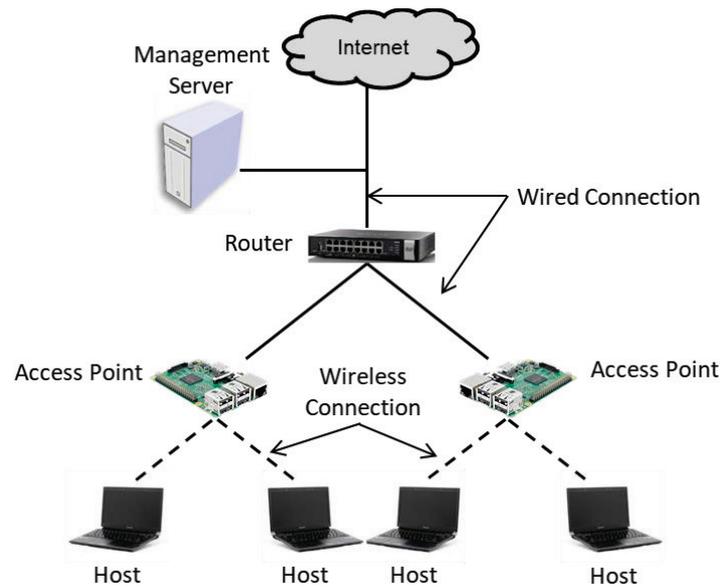


Figure 4. Example topology of an elastic WLAN system.

3.3. Active AP Configuration Algorithm for Dual Interfaces

The *active AP configuration algorithm* selects the active APs and their host associations that can minimize the number of active APs with dual interfaces under the *minimum host throughput constraint*.

Figure 5 shows the flow chart for the active AP configuration algorithm for the dual interface.

3.3.1. Formulation

The problem formulation for this algorithm is formulated as follows:

1. **Inputs:**

- number of hosts: H ,
- number of APs: N ,
- estimated throughput between AP_i and $host_j$ for $i = 1$ to N , $j = 1$ to H at each interface: tp_{ij} ,
- minimum throughput for the association: S ,
- number of orthogonal channels (OCs) for each interface: C ,
- minimum host throughput: G ,
- available total throughput: B^a .

2. **Outputs:**

- set of active APs with dual interfaces,
- set of hosts associated with each interface at every active AP,
- channel assigned to each interface at every active AP.

3. **Objectives:**

- E_1 represents the number of active access points (APs) with the dual interfaces to be minimized under the *minimum host throughput constraint*:

$$E_1 = [\text{number of active APs}] \tag{6}$$

- holding to the first objective, to maximize the *minimum average host throughput* E_2 :

$$E_2 = \min_j [TH_j] \tag{7}$$

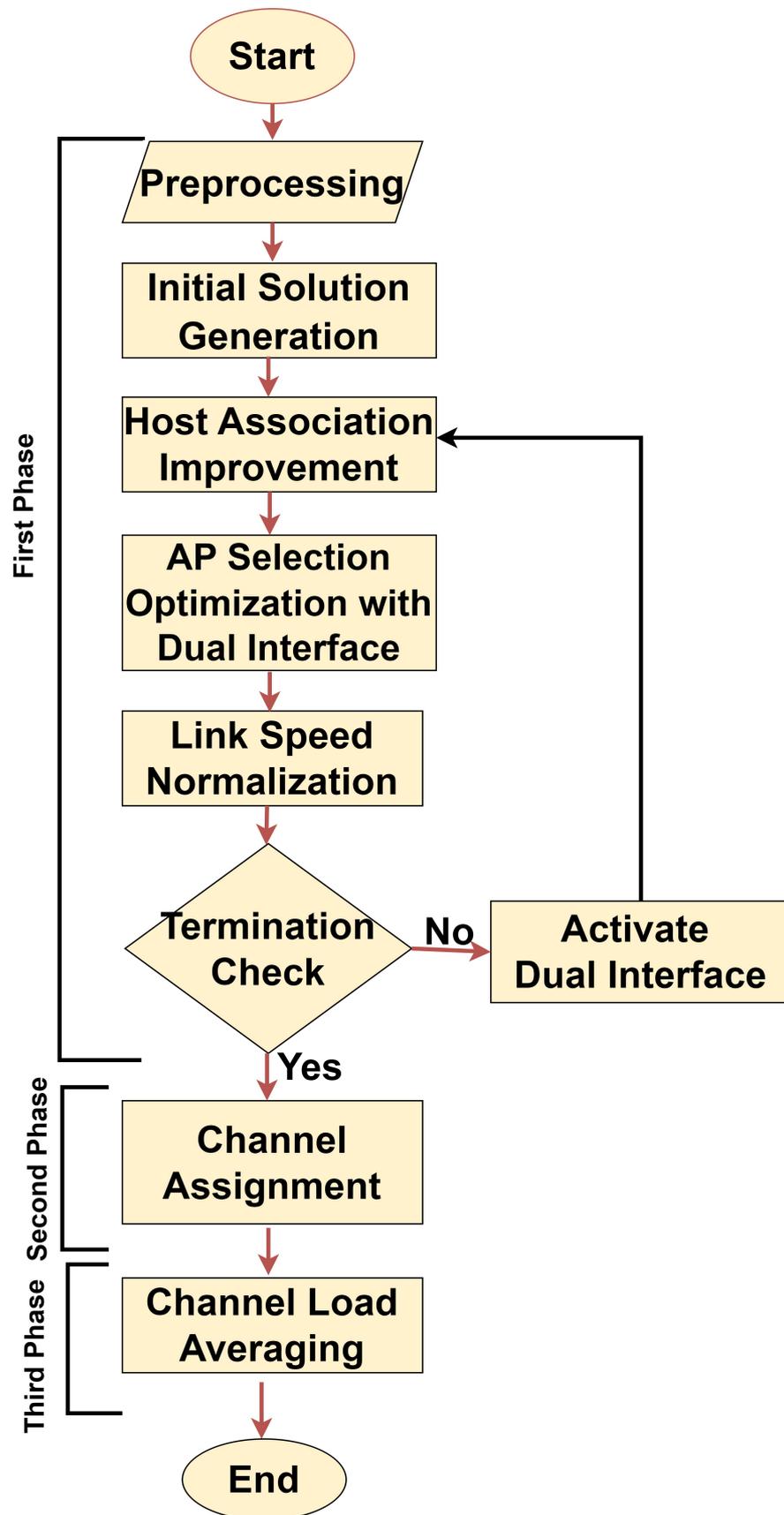


Figure 5. Flowchart of dual AP configuration algorithm.

where TH_j represents the average host throughput for AP_j that is given by:

$$TH_j = \frac{1}{\sum_k \frac{1}{tp}} \quad (8)$$

where tp represents the link speed between $node_j$ and $node_k$ ($link_{jk}$).

- holding to the two objectives, to minimize the *total interfered communication time* E_3 for channel assignments:

$$E_3 = \sum_{i=1}^N \left[\sum_{\substack{k \in I_i \\ c_k = c_i}} T_k \right] \quad (9)$$

where T_k is the AP_i total communication time, I_i represents the set of the APs interfered with AP_i , and c_i indicates the channel assigned to AP_i .

4. Constraints:

- **minimum host throughput:** the throughput of any host must be larger than or equal to the *minimum host throughput* G on average when all the hosts are communicating simultaneously.
- **total throughput:** the total throughputs for all the hosts must be smaller than or equal to the *available total throughput* B^a .
- **channel assignment:** each interface of an AP must be assigned a channel.

3.3.2. Algorithm Procedure

The algorithm consists of three phases. The procedure is described as follows:

1. **First Phase:** The first phase of the algorithm selects the active APs with the *dual interface* and their host associations to minimize E_1 and maximize E_2 [33].
 - (a) **Preprocessing:** Initially, the APs and the hosts locations are used as inputs of the algorithm. The AP locations are selected manually in the network, taking into account the electrical power supply, the coverage, and user demands. The throughput is estimated using the *throughput estimation model* in Equation (3) for every possible pair of AP and host. In addition, the 11n interface of an AP is chosen as the initial candidate one for any host.
 - (b) **Initial Solution Generation:** *Greedy algorithm* is adopted to calculate the initial solution E_1 [51].
 - (c) **Host Association Improvement:** The minimum host throughput and the overall throughput in the network are improved by randomly changing the host association according to the procedure in [22].
 - (d) **AP Selection Optimization:** This phase optimizes the selection of number of active dual interfaces APs and the AP-host associations in order to further minimize both E_1 and E_2 using the *local search method* [52].
 - (e) **Link Speed Normalization:** The fairness criterion will be used if the total expected bandwidth is greater than B^a . Next, the speed of the link has been normalized.
 - (f) **Termination Check:** If either of the two interfaces for each current AP is not activated, then activate the interface and apply the *host association improvement* phase. After that, the algorithm will be terminated and proceed to phase 2 if the minimum throughput constraint of the host is satisfied. If not, proceed to the *AP selection optimization* phase.
2. **Second Phase:** The second phase assigns a channel to each active AP interface to minimize E_3 [22].

- (a) **Preprocessing:** The interference and delay conditions of the network are illustrated by a graph.
 - (b) **Interfered AP Set Generation:** The set of interfering AP interfaces is found for each AP interface.
 - (c) **Initial Solution Construction:** *The greedy algorithm* is adopted to calculate the the initial solution.
 - (d) **Solution Improvement by Simulated Annealing:** Simulated annealing (SA) is the probabilistic optimization technique that can be used to improve the solution by gradually adjusting the solution over time. In our proposal, the SA is used to optimize the channel assignment for each interface of every active AP to improve the performance of the network. The SA procedure is used with the constant *temperature* T^{SA} for the *repeating times* R^{SA} , where T^{SA} and R^{SA} are given as the algorithm parameters.
3. **Third Phase:** The third phase averages the loads among the different channels in order to minimize E_3 [22].
- (a) **Initialization:** The AP flag is initialized with 0(= OFF) in every AP. This flag is used to prevent the re-processing of the same AP again.
 - (b) **AP Selection:** One OFF flag AP is chosen to move its associated host to a different AP to which a different channel is assigned.
 - (c) **Host Selection:** One associated host with the selected AP is selected to perform the AP movement.
 - (d) **Change Application:** Finally, the new associated AP is selected for that host.

4. Interface Selection

In this section, we examine the two cases of interface selections through experiments. In addition, we introduce the throughput reduction factor to improve the estimation accuracy under higher interference and examine the estimation accuracy in the experimental results.

4.1. Experimental Setup for Interface Selection

Raspberry Pi 4B is a small, low-powered, card-sized, and single-board computer. It can be used in a variety of practical systems requiring some computation or networking abilities. It has a built-in wireless network interface (NIC) that supports IEEE 802.11n and 11ac. By using *hostapd* software, it can be used as a *software AP* in WLAN [53].

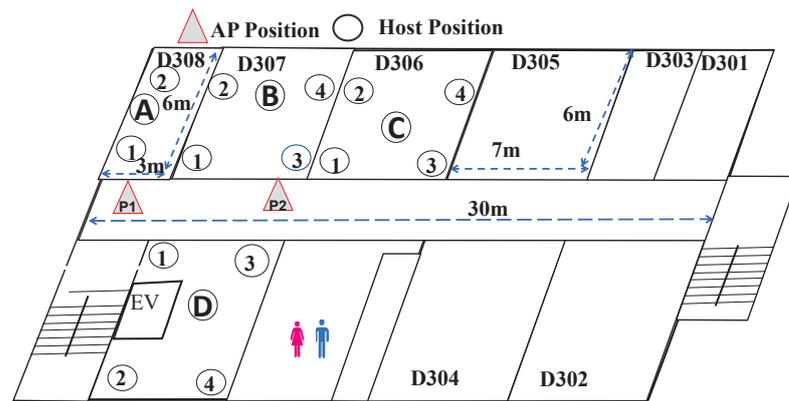
Archer T4U is a wireless adapter and is designed to work with the latest wireless standards, including IEEE 802.11ac, to allow faster data transfers and wider ranges. It has a high-performance antenna that is designed to improve the signal strength and reduce interference. Therefore, we can realize the AP that has two network interfaces with a low cost.

In WLAN, the throughput performance can be different by the protocol selections of the embedded NIC and the external one. We consider the two groups for *Raspberry Pi* and *Archer T4u* adapter for measurements as follows:

Group A: Raspberry Pi for 11n at 2.4 GHz and external for 11ac at 5 GHz.

Group B: Raspberry Pi for 11ac at 5 GHz and external for 11n at 2.4 GHz.

Firstly, we conducted experiments by measuring the *single link throughput* for two network topologies. Figure 6 illustrates the experimental field on the 3rd floor of Engineering Building #2 at Okayama University. The host locations are represented by circles, and the AP locations are represented by triangles. The 40 MHz bonded channel is used for both interfaces. The dual interfaces were configured through the commands in [33].



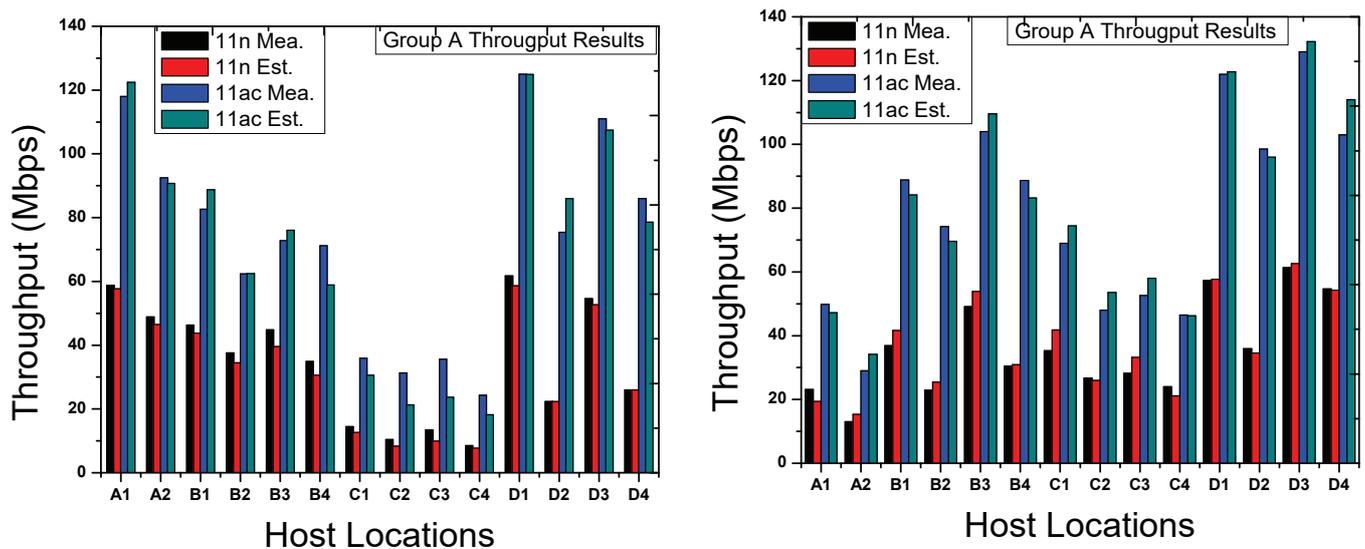
3rd Floor Engineering Building #2, Okayama University

Figure 6. Experiment field.

4.2. Throughput Results

Figures 7 and 8 show the measured and estimated single-link throughput results for 40 MHz for Group A and Group B. In each Figure, 11n Mea. and 11ac Mea. represent the measured throughput for the 11n and 11ac interface; 11n Est. and 11ac Est. represent the estimated throughput for the 11n and 11ac interface. For both interfaces, the throughput is estimated using Equation (3) by the parameter optimization tool in Table 2.

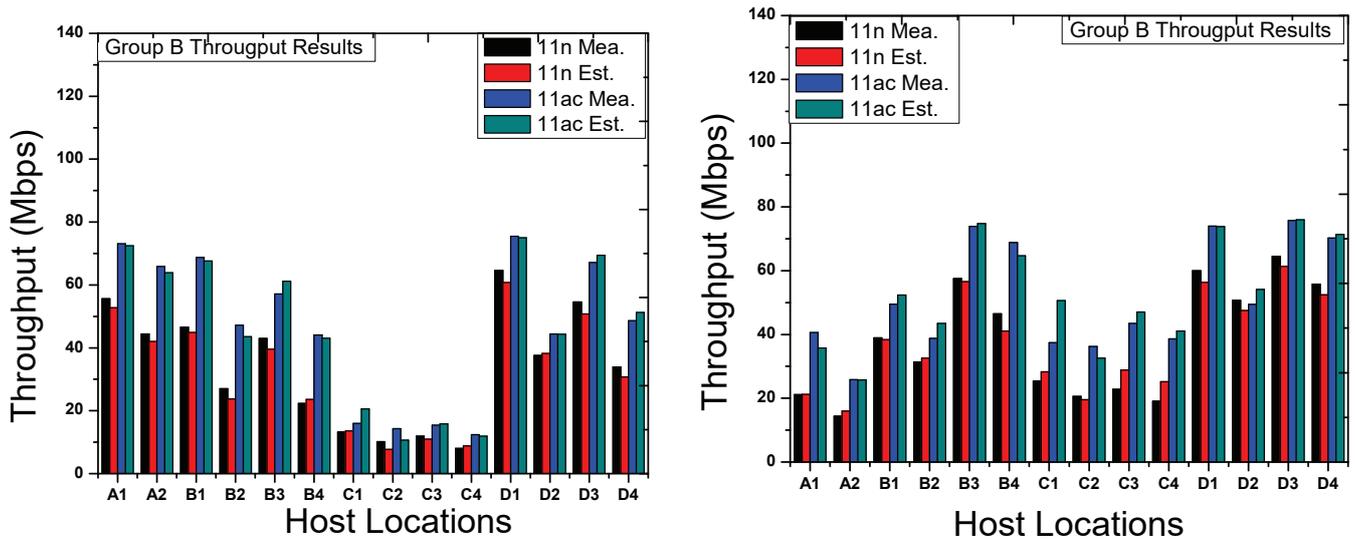
To obtain more accurate results, we choose 14 different host locations in the experimental field with two different AP positions. Figures 7 and 8 show that Group A is better than Group B for the dual interface, and the measured throughput and the estimated throughput matched well for each host position. In Group A, Raspberry Pi is used for 11n and Archer T4U is 11ac, and in Group B, they are opposites. The performance of Group A is better than that of Group B, because Archer T4U used the more advanced wireless chipset and a higher-quality antenna than Raspberry Pi, which can enhance the throughput performance when Archer T4U is used for the faster communication protocol of 11ac at 5 GHz. Thus, we should use the built-in wireless network interface card (NIC) of the Raspberry Pi device for 11n, and the Archer T4U wireless NIC adapter is for 11ac.



(a) Results for AP1.

(b) Results for AP2.

Figure 7. Group A throughput results for 40 MHz.



(a) Results for AP1.

(b) Results for AP2.

Figure 8. Group B throughput results for 40 MHz.

5. Modification of Active AP Configuration Algorithm

In this section, we modify the active AP configuration algorithm for *dual interfaces* with the *channel bonding (CB)*.

5.1. Modifications of the Problem Formulation

The *objectives* of the problem formulation for the algorithm is modified to consider CB as follows:

1. **Objectives:**

The average host throughput calculation of AP_j in Equation (8) is modified as follows:

$$TH_j = \frac{1}{\sum_k \frac{1}{tp \times srf(m)}} \tag{10}$$

5.2. Extension of the Algorithm's Procedure

Host association improvement in the first phase of the algorithm procedure is modified:

1. Change the association of each host to the interface of AP which provides the highest throughput using Equation (4) within the associable AP interfaces. At this stage, compute the cost function E_2 and keep it as the best found cost function, E_2^{best} .
2. Find the interface of AP that provides the lowest host throughput using Equation (7). Then, make the *modifiable hosts* list that are associated with the AP interface, which can be connected with another AP interfaces.
3. Choose one host at random from the list of *modifiable hosts*. Then, this host is associated at random with other associable active AP interface. Compute the new cost function E_2^{new} .
4. Replace E_2^{best} with the newly found E_2^{new} , if $E_2^{new} > E_2^{best}$, and retain the new AP-host association. If not, go back to the previous association and select the best cost function E_2^{best} .

6. Evaluations

In this section, we evaluate the proposal through simulations using the *WIMNET* simulator and experiments using the testbed system.

6.1. Simulation Setup

Originally, the *WIMNET simulator* [36] was developed to evaluate the performance of a large-scale wireless Internet-access mesh network on a standard PC with a reasonable CPU time. The simulator has been modified to be used in conventional WLAN simulations with multiple access points. It has been used to simulate various WLANs, including different topologies, channel models, and interference conditions. Tables 3 and 4 show the hardware and software specifications in the simulations using the *WIMNET simulator*.

Table 3. PC environment.

simulator	WIMNET
CPU	Intel Core i7
memory	8 GB
OS	Ubuntu LTS 14.04

Table 4. Simulation parameters for the *WIMNET simulator*.

Parameter	Values
packet size	1500 bytes
max. transmission rate	150 Mbit/s
propagation model	log distance path loss model
rate adaptation algorithm	link speed estimation model [24]
carrier sense threshold	−85 dBm
transmission power	19 dBm
collision threshold	10
RTS/CTS	yes

6.2. Experimental Setup

In the experiments, *Raspberry Pi 4B* is used for each access point (AP) by running *hostapd* [54]. The embedded wireless NIC is used for the 11n interface, and the *Archer T4U* wireless NIC adapter is used for the 11ac interface. The *Linux* laptop PCs are used for the client hosts and server. For both interfaces, the 40 MHz bonded channel is used. The hardware and software presented in Table 5 are taken into account in the measurements.

To set the bonded channel for both interfaces, we can modify the `/etc/hostapd/hostapd.conf` file using the following Linux commands:

```
$ #40 MHz for 11n
$ rsn_pairwise=CCMP
$ ht_capab=[HT40+][SHORT-GI-20][SHORT-GI-40][DSSS_CCK-40]
[MAX-AMSDU-3839]
```

```
$ #40 MHz for 11ac
$ rsn_pairwise=CCMP
$ vht_capab=[MAX-MPDU-3895][SHORT-GI-80][SU-BEAMFORMEE]
```

The `[HT40+]` needs to set the 20 MHz primary channel in the `/etc/hostapd/hostapd.conf` file. Then, the primary channel will be bonded with the secondary channel to form the 40 MHz channel.

iperf 2.0.5 [55] software is adopted for the throughput measurements as a popular tool for measuring the TCP throughput by sending TCP packets from the server to the host. The TCP traffic at a link automatically saturates by this software. The TCP traffic was generated for each measurement point in this paper using a 477 KB TCP window size and an 8 KB buffer size for four minutes at intervals of 30 s, which was conducted twice on different days. Then, the average value is calculated for each point and is shown in this paper. The packet size of the generated TCP traffic is 1500 bytes. For each point, the average of the measured throughput is computed over four minutes at 30 s intervals on two separate days in order to eliminate the fluctuations in measurements.

The elastic WLAN testbed system using *Raspberry Pi* APs with single interfaces was implemented and used in [13,14,22,56–58]. We extend this testbed by adding *Archer T4U* at the AP for the dual interface, as shown in Figure 9. The testbed included four Linux laptop PCs for the hosts and one Linux laptop PC for the server that is used to collect the necessary information for the active AP configuration algorithm and controls the APs and the hosts. This server controls the overall system through the following steps:

1. The server obtains active all the APs in the network.
2. It collects the information of the connected devices in the network using *arp-scan-tool* in Linux, which includes the IP and MAC addresses of the APs and the hosts.
3. It collects the *receiving signal strength (RSS)* at each host from each AP using *nm-tool*.
4. It converts the RSS to the throughput using the *sigmoid function* in [23].
5. It executes the *active AP configuration algorithm* using the inputs derived in the previous steps and obtains the algorithm output that consists of the number of active APs, the host associations, and the channels assigned to the active APs.
6. It activates or deactivates the APs in the network according to the algorithm output using the following command.
\$ sudo /etc/init.d/hostapd start
\$ sudo /etc/init.d/hostapd stop
7. It changes the associations of the hosts with the APs according to the algorithm output using *nmcli-tool*.
8. It assigns the channels to the APs according to the algorithm output using *sed-tool*.

Table 5. Devices and software specifications.

Access Point	
model	Raspberry Pi 4B
CPU	Broadcom BCM2711 @1.5 GHz
RAM	8 GB LPDDR4-3200 SDRAM
Operating System	Linux Raspbian
Software	hostapd V 2.10
External NIC	Archer T4U V3.0 AC1300
server PC	
Model	Fujitsu Lifebook S761/C
CPU	Intel Core i5-2520M @2.5 GHz
RAM	4 GB DDR3 1333 MHz
Operating System	Linux Ubuntu 14.04 LTS (kernel 3.13.0-57)
Software	iperf 2.0.5
host PC	
Model	1. Toshiba Dynabook R731/B 2. Toshiba Dynabook R734/K 3. Fujitsu Lifebook S761/C
CPU	1. Intel Core i5-2520M @2.5 GHz 2. Intel Core i5-4300M @2.6 GHz 3. Intel Core i5-2520M @2.5 GHz
RAM	4 GB DDR3 1333MHz
Operating System	Linux Ubuntu 14.04 LTS (kernel 3.13.0-57)
Software	iperf 2.0.5

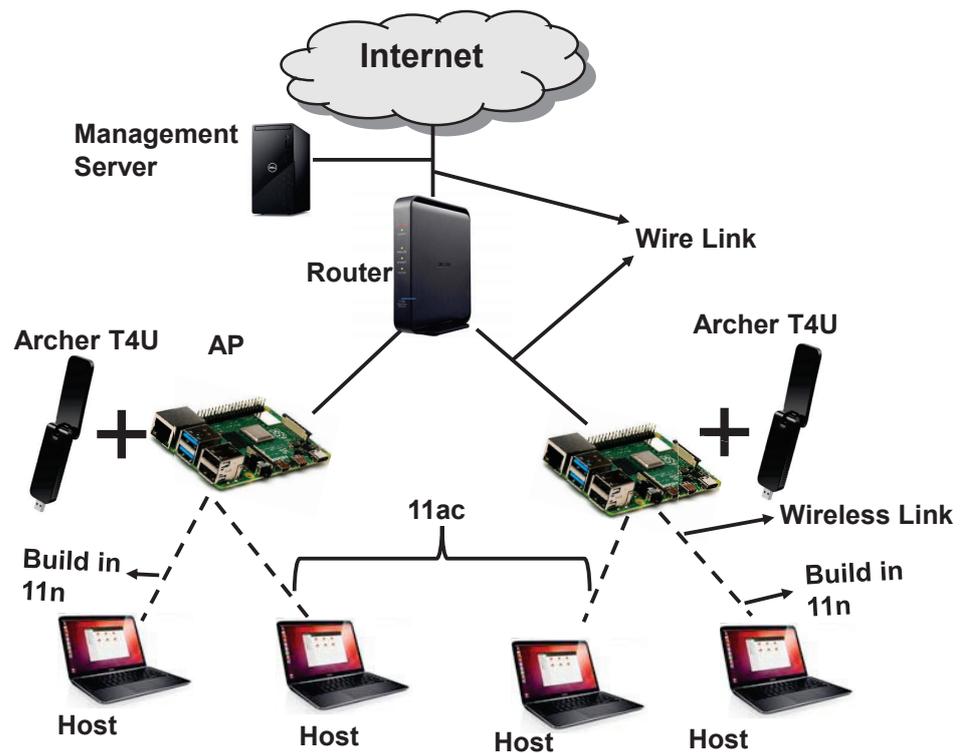


Figure 9. Testbed topology for dual interface.

6.3. Network Fields

The 3rd floor of Engineering Building #2 and the 2nd floor of Graduate School Building at Okayama University were considered as the network fields for simulations and experiments. Figures 2 and 3 illustrate the network topologies with the AP and host locations in the two fields, respectively. For each topology, we considered five APs with ten hosts. The two different network environments were chosen to show the results in different environments. In both fields, several other WLANs were observed, which may have caused interference in our experiments. In the daytime, a large number of wireless devices, such as smartphones, laptops, and IoT devices, transmit wireless signals. Thus, the daytime situation will make it difficult to measure the performance of the WLAN accurately. The signals from other devices can interfere with the signals being transmitted by the devices under testing, which can result in inaccurate throughput measurements. At night, fewer wireless devices will transmit signals. Thus, the level of interference can be reduced and will make it easier to accurately measure the performance of a WLAN.

6.4. Results and Discussions

Table 6 shows the simulation and measurement results for the two fields. The *min. host thro. cons. G* (Mbps) represents the threshold value for the *minimum host throughput constraint*. This value is used as the benchmark. All the hosts in the network should achieve it regardless of the relative distances from the associated APs. In our study, the value was selected manually, taking into account the number of hosts, the number of APs, and the overall network capacity of the WLAN. The number of active APs should be increased when this constant is not satisfied. In Table 6, the *ave. min. host thro.* (Mbps) represents the average minimum host throughput (Mbps), and the *ave. overall thro.* (Mbps) represents the average overall throughput in the network. The *sim.* represents the simulation results, and *mea.* represents the measurement results.

Note: *field #1* in Table 6 corresponds to Engineering Building #2. The *minimum host throughput constraint* is applied with $G = 10$ Mbps and $G = 25$ Mbps. The throughput results from the simulation are well matched with the measurement results in any topology.

This confirms the accuracy of the proposed algorithm, and the use of the dual interface at the AP can minimize the number of active APs.

In addition, *field #2* in Table 6 corresponds to *Graduate School Building*. The same *minimum host throughput constraint* is considered in this field. Again, the throughput results by the simulation are well matched with the measurement results in each topology. Thus, the accuracy and effectiveness of the proposed algorithm are confirmed.

Additionally, we evaluated the proposal by comparing the performances between the single 11ac interface case and the dual interface case when the same number of APs are activated. As shown in Table 7, both the minimum host throughput and the average overall throughput results were increased by the proposal. The dual interfaces at the AP operated at different frequency bands. It allows one AP to communicate with two hosts simultaneously without interference. In addition, it can reduce interference between neighboring APs in a dense WLAN, because the number of active APs can be reduced. As a result, the performance of the proposal is greater than that of the single 11ac interface case. The measured throughput results are well-matched with the simulated ones in any case. It is confirmed that the proposal provides higher overall throughput than 11ac only when the same number of APs are activated.

Table 6. Simulation and measurement results for CB cases.

Exp. Field	Topology	Min. Host Thro. Cons. G (Mbps)	Single Interface						Dual Interface		
			11n			11ac			Total Active APs	Ave. Min. Host Thro. (Mbps)	
			Total Active APs	Ave. Min. Host Thro. (Mbps)		Total Active APs	Ave. Min. Host Thro. (Mbps)			both	sim.
				both	sim.		mea.	both	sim.		
field #1	1	10	3	13.32	12.56	2	14.78	13.66	1	12.32	11.47
		25	5	26.34	25.02	3	26.64	25.08	2	26.65	25.19
	2	10	3	13.58	12.77	2	16.54	14.84	1	13.09	12.19
		25	5	25.13	24.78	3	26.23	25.09	2	26.89	25.37
field #2	3	10	3	12.93	11.76	2	12.50	11.36	1	12.34	11.53
		25	5	25.01	24.04	4	31.18	29.63	3	37.71	35.83
	4	10	3	13.45	12.66	3	17.39	15.73	2	19.94	18.25
		25	5	23.80	22.85	4	27.71	26.02	3	36.55	34.76

Table 7. Comparative analysis between 11ac and the dual interface.

Exp. Field	Topology	Minimum Host Throughput Constraint G (Mbps)	Dual Interface				
			Total Active APs	Average Minimum Host Throughput (Mbps)		Average Overall Throughput (Mbps)	
				both	sim.	mea.	sim.
field #1	1	20	2	26.65	25.19	267.19	261.93
		30	3	37.88	35.92	235.24	231.06
	2	20	2	26.89	25.37	259.12	255.97
		30	3	36.38	34.02	225.77	220.63
field #2	3	20	2	24.22	22.32	266.34	259.36
		40	4	45.72	42.36	215.75	210.36
	4	20	3	36.55	34.76	238.45	233.93
		40	4	43.67	40.92	222.35	217.21

6.5. Comparisons with Previous Studies

To clarify the effectiveness of the proposed extended algorithm, we conducted simulations and experiments using the same network configuration as that for our previous algorithm [33].

Table 8 shows the simulation and measurement results for both the single interface and the dual interface under non-CB cases in both experimental fields when the same number of APs was activated as before. The results in Table 8 show that the average minimum host throughput and the average overall throughput results become smaller due to the smaller channel bandwidth, compared with the proposal. Figures 10–13 show the average overall throughput with respect to the number of active APs for each topology for both CB and non-CB cases. It is noted that the simulated results are well-matched with the measured ones for all the cases. This confirms that the proposal with the CB provides higher throughput than the previous one with the non-CB, regardless of the higher interference.

Table 8. Simulation and measurement results for non-CB cases [33].

Exp. Field	Topology	Min. Host Thro. Cons. G (Mbps)	Single Interface						Dual Interface		
			11n			11ac			Total Active APs	Ave. Min. Host Thro. (Mbps)	
			Total Active APs	Ave. Min. Host Thro. (Mbps)		Total Active APs	Ave. Min. Host Thro. (Mbps)			both	sim.
			both	sim.	mea.	both	sim.	mea.			
field #1	1	5	3	8.20	6.96	2	8.78	7.14	1	8.14	7.82
		15	5	14.05	12.26	3	15.12	13.25	2	16.11	15.16
	2	5	3	8.15	6.93	2	9.21	7.96	1	8.14	7.02
		15	5	14.46	12.98	3	15.65	14.42	2	16.43	14.88
field #2	3	5	3	7.95	6.90	2	9.21	8.11	1	8.01	7.12
		15	5	15.05	13.81	4	17.79	16.08	3	19.95	17.96
	4	5	3	8.34	7.07	3	12.89	11.06	2	13.24	12.10
		15	5	15.97	13.89	4	19.04	17.35	3	21.15	20.67

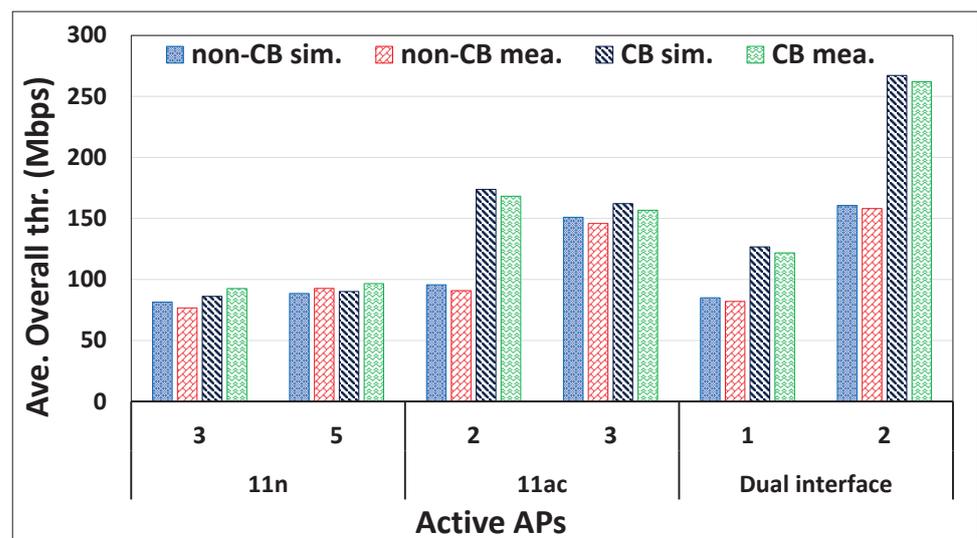


Figure 10. Throughput results for Topology 1.

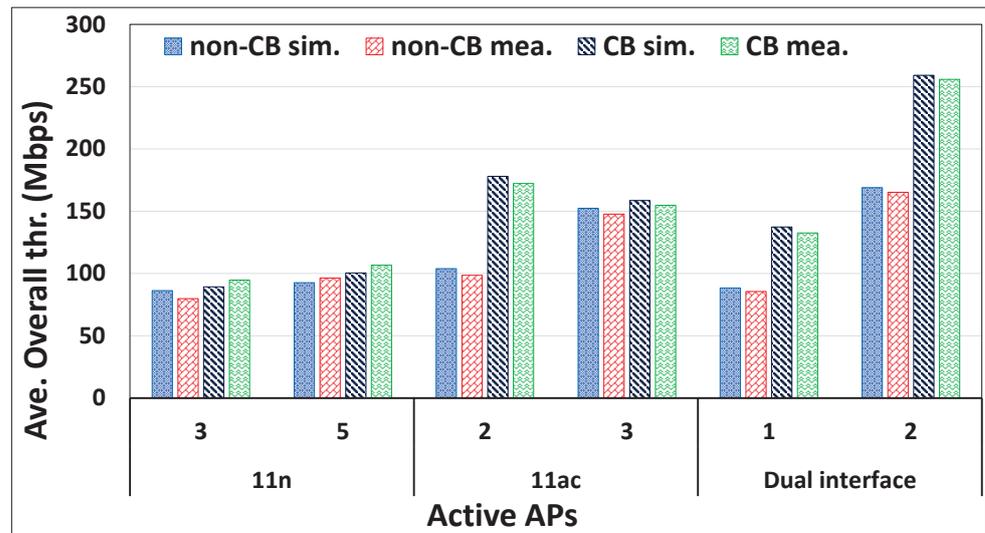


Figure 11. Throughput results for Topology 2.

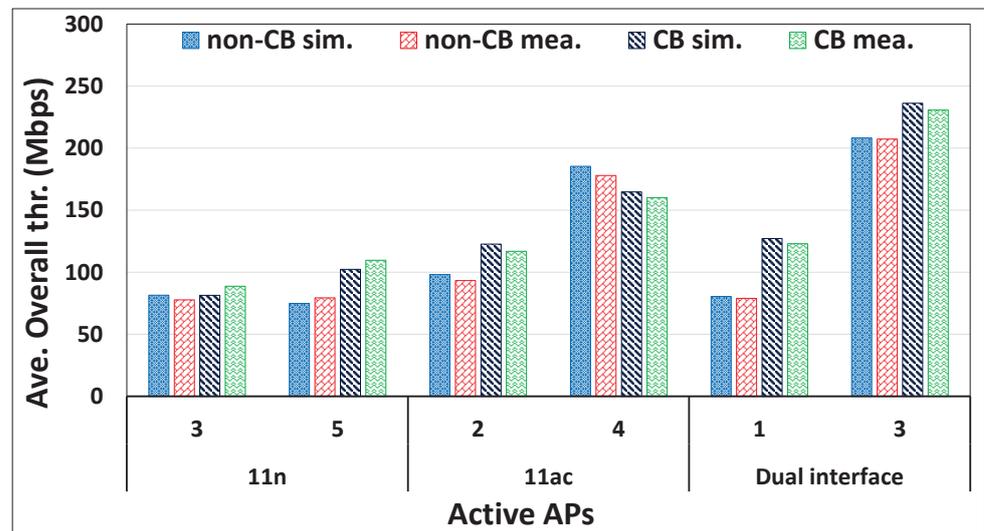


Figure 12. Throughput results for Topology 3.

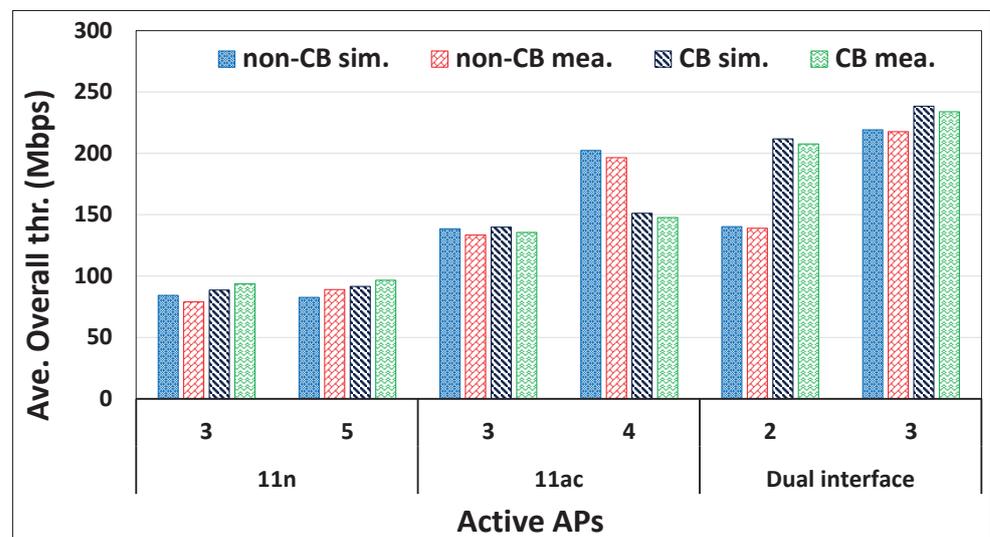


Figure 13. Throughput results for Topology 4.

To compare the throughput performance of our approach with that of an existing approach, the *band-steering (BSTR)* method using the dual interfaces [59] was considered in experiments. In this method, each host is associated with the AP that provides the highest *received signal strength indicator (RSSI)* among the active APs.

The same number of active APs with dual interfaces and the same number of hosts are used as the ones for the proposal. Figures 14 and 15 show the average minimum host throughput and the average overall throughput results with respect to the number of active APs in *Engineering Building #2 at Okayama University* as the network field by *BSTR* and by the proposal. The results suggest that the proposal provides higher throughput than *BSTR* in any case. Thus, the effectiveness of the proposal has been confirmed.

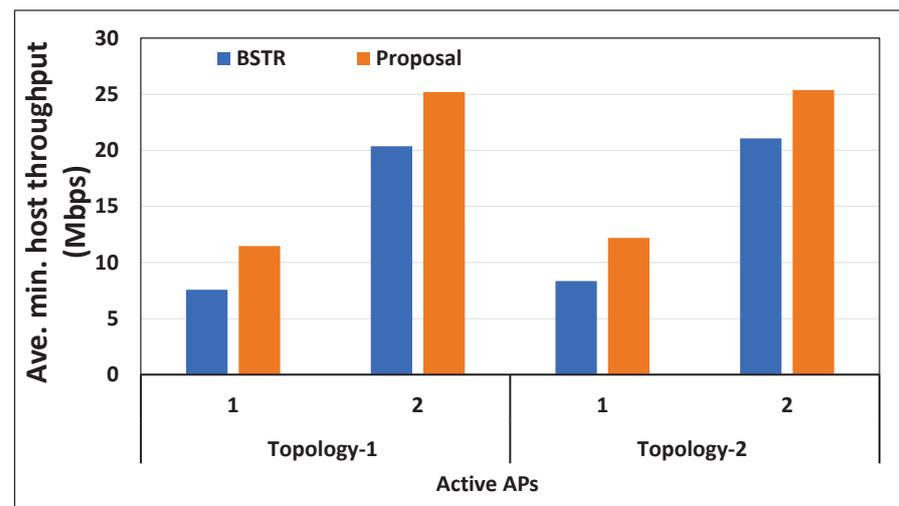


Figure 14. Minimum host throughput result.

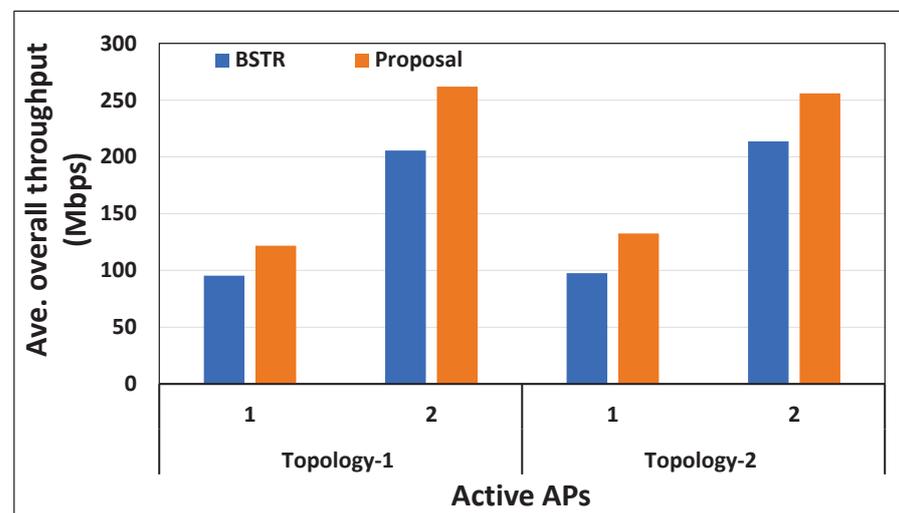


Figure 15. Overall throughput result.

7. Conclusions

This paper studied the *active access-point (AP) configuration algorithm* for the *IEEE 802.11n/ac wireless local-area network (WLAN)* with *dual network interfaces* and *channel bonding* for the APs. The main goal of this algorithm is to optimize the number of active APs where each host satisfies the minimum host throughput constraint in the network. By reducing the number of active APs, it also aims to reduce the amount of energy consumed by the network. The effectiveness of the proposed algorithm was confirmed through extensive simulations using the *WIMNET* simulator and testbed experiments using *Raspberry Pi* with

the external interface in two different network fields. In future works, we will further improve the AP configuration algorithm by considering partially overlapping channels, transmission power optimizations, and the 11ax protocol. Then, we will evaluate our proposals in various network fields and topologies to confirm their effectiveness.

Author Contributions: Conceptualization, S.C.R. and N.F.; methodology, S.C.R. and N.F.; software, S.C.R.; validation, S.C.R., M.M.R. and B.W.; formal analysis, S.C.R.; investigation, S.C.R.; resources, S.C.R.; data curation, S.C.R., M.M.R. and B.W.; writing—original draft preparation, S.C.R.; writing—review and editing, S.C.R., N.F., M.K. and W.-C.K.; visualization, S.C.R.; supervision, N.F.; project administration, N.F.; funding acquisition, N.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sadhu, P.K.; Yanambaka, V.P.; Abdelgawad, A. Internet of Things: Security and solutions survey. *Sensors* **2022**, *22*, 7433. [[CrossRef](#)] [[PubMed](#)]
2. Ahsan, M.; Teay, S.H.; Sayem, A.S.M.; Albarbar, A. Smart clothing framework for health monitoring applications. *Signals* **2022**, *3*, 113–145. [[CrossRef](#)]
3. Ahmad, K.; Maabreh, M.; Ghaly, M.; Khan, K.; Qadir, J.; Al-Fuqaha, A. Developing future human-centered smart cities: Critical analysis of smart city security, data management, and ethical challenges. *Comput. Sci. Rev.* **2022**, *43*, 100452. [[CrossRef](#)]
4. Sinha, B.B.; Dhanalakshmi, R. Recent advancements and challenges of Internet of Things in smart agriculture: A survey. *Future Gener. Comput. Syst.* **2022**, *126*, 169–184. [[CrossRef](#)]
5. Mohamad Jawad, H.H.; Bin Hassan, Z.; Zaidan, B.B.; Mohammed Jawad, F.H.; Mohamed Jawad, D.H.; Alredany, W.H.D. A systematic literature review of enabling IoT in healthcare: Motivations, challenges, and recommendations. *Electronics* **2022**, *11*, 3223. [[CrossRef](#)]
6. Mocrii, D.; Chen, Y.; Musilek, P. IoT-based smart homes: A review of system architecture, software, communications, privacy and security. *Int. Things* **2018**, *1*, 81–98. [[CrossRef](#)]
7. Agyemang, J.O.; Kponyo, J.J.; Klogo, G.S.; Boateng, J.O. Lightweight rogue access point detection algorithm for WiFi-enabled Internet of Things (IoT) devices. *Int. Things* **2020**, *11*, 100200. [[CrossRef](#)]
8. Munene, K.I.; Funabiki, N.; Rahman, M.M.; Briantoro, H.; Roy, S.C.; Kuribayashi, M. A throughput drop estimation model and its application to joint optimization of transmission power, frequency channel, and channel bonding in IEEE 802.11 n WLAN for large-scale IoT environments. *Int. Things* **2022**, *20*, 100583. [[CrossRef](#)]
9. Brito, T.; Pereira, A.I.; Lima, J.; Valente, A. Wireless sensor network for ignitions detection: An IoT approach. *Electronics* **2020**, *9*, 893. [[CrossRef](#)]
10. Gupta, B.B.; Quamara, M. An overview of Internet of Things (IoT): Architectural aspects, challenges, and protocols. *Concurr. Comput. Pract. Exper.* **2020**, *32*, e4946. [[CrossRef](#)]
11. Prasad, S.B.; Madhumathy, P. Long term evolution for secured smart railway communications using internet of things. In *Machine Learning Algorithms for Industrial Applications*; Das, S., Das, S., Dey, N., Hassanien, A.E., Eds.; Springer: Cham, Switzerland, 2021; pp. 285–300, ISBN 978-3-030-50641-4.
12. Hong, H.; Kim, Y.Y.; Kim, R.Y. A low-power WLAN communication scheme for IoT WLAN devices using wake-up receivers. *Appl. Sci.* **2018**, *8*, 72. [[CrossRef](#)]
13. Islam, M.M.; Funabiki, N.; Sudiby, R.W.; Munene, K.I.; Kao, W.C. A dynamic access-point transmission power minimization method using PI feedback control in elastic WLAN system for IoT applications. *Int. Things* **2019**, *8*, 100089. [[CrossRef](#)]
14. Islam, M.M. A Study of Dynamic Access-Point Configuration and Power Minimization in Elastic Wireless Local-Area Network System. Ph.D. Thesis, Graduate School of Natural Science and Technology, Okayama University, Okayama, Japan, 2019.
15. Crow, B.P.; Widjaja, I.; Kim, J.G.; Sakai, P.T. IEEE 802.11 wireless local area networks. *IEEE Commun. Mag.* **1997**, *35*, 116–126. [[CrossRef](#)]
16. Gast, M. *802.11 n: A Survival Guide*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2012; ISBN 9781449312046.
17. Kassa, L.; Davis, M.; Cai, J.; Deng, J. A new adaptive frame aggregation method for downlink WLAN MU-MIMO channels. *J. Commun.* **2021**, *16*, 311–322. [[CrossRef](#)]
18. Kang, S.L.; Chen, G.Y.H.; Rogers, J. Wireless LAN access point location planning. In Proceedings of the Institute of Industrial Engineers Asian Conference, Taipei, Taiwan, 18–20 July 2013; pp. 907–914.
19. Al Mamun, M.S.; Islam, M.E.; Funabiki, N.; Kuribayashi, M.; Lai, I.W. An active access-point configuration algorithm for elastic wireless local-area network system using heterogeneous devices. *Int. J. Netw. Comput.* **2016**, *6*, 395–419.

20. Kim, Y.; Kim, M.S.; Lee, S.; Griffith, D.; Golmie, N. AP selection algorithm with adaptive CCAT for dense wireless networks. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), San Francisco, CA, USA, 19–22 March 2017; pp. 1–6.
21. Munene, K.I.; Funabiki, N.; Islam, M.M.; Kuribayashi, M.; Al Mamun, M.S.; Kao, W.C. An extension of throughput drop estimation model for three-link concurrent communications under partially overlapping channels and channel bonding in IEEE 802.11 n WLAN. *Adv. Sci. Technol. Eng. Syst. J.* **2019**, *4*, 94–105. [[CrossRef](#)]
22. Al Mamun, M.S.; Funabiki, N.; Lwin, K.S.; Islam, M.E.; Kao, W.C. A channel assignment extension of active access-point configuration algorithm for elastic WLAN system and its implementation using Raspberry Pi. *Int. J. Netw. Comput.* **2017**, *7*, 248–270. [[CrossRef](#)]
23. Funabiki, N.; Taniguchi, C.; Lwin, K.S.; Zaw, K.K.; Kao, W.C. A parameter optimization tool and its application to throughput estimation model for wireless LAN. In Proceedings of the Conference on Complex, Intelligent, and Software Intensive Systems, Turin, Italy, 10–13 July 2017; pp. 701–710.
24. Lwin, K.S.; Funabiki, N.; Taniguchi, C.; Zaw, K.K.; Al Mamun, M.S.; Kuribayashi, M.; Kao, W.C. A minimax approach for access point setup optimization in IEEE 802.11 n wireless networks. *Int. J. Netw. Comput.* **2017**, *7*, 187–207.
25. Deek, L.; Garcia-Villegas, E.; Belding, E.; Lee, S.J.; Almeroth, K. The impact of channel bonding on 802.11 n network management. In Proceedings of the Seventh Conference on emerging Networking EXperiments and Technologies, Tokyo, Japan, 6–9 December 2011; pp. 1–12.
26. Han, M.; Khairy, S.; Cai, L.X.; Cheng, Y. Performance analysis of opportunistic channel bonding in multi-channel WLANs. In Proceedings of the IEEE Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016; pp. 1–6.
27. Gast, M.S. *802.11 ac: A Survival Guide: Wi-Fi at Gigabit and Beyond*, 1st ed.; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2013.
28. Daldoul, Y.; Meddour, D.E.; Ksentini, A. IEEE 802.11 ac: Effect of channel bonding on spectrum utilization in dense environments. In Proceedings of the IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017; pp. 1–6.
29. Khorov, E.; Kiryanov, A.; Lyakhov, A.; Bianchi, G. A tutorial on IEEE 802.11 ax high efficiency WLANs. *IEEE Commun. Surv. Tuts.* **2019**, *21*, 197–216. [[CrossRef](#)]
30. BUFFALO WI-U3-1200AX2. Available online: <https://www.buffalo.jp/product/detail/wi-u3-1200ax2.html> (accessed on 21 February 2023).
31. National Instrument. Introduction to Wireless LAN Measurements from 802.11a to 802.11ac. 2018. Available online: http://download.ni.com/evaluation/rf/Introduction_to_WLAN_Testing.pdf (accessed on 24 November 2022).
32. Barrachina-Muñoz, S.; Wilhelmi, F.; Bellalta, B. To overlap or not to overlap: Enabling channel bonding in high-density WLANs. *Comput. Netw.* **2019**, *152*, 40–53. [[CrossRef](#)]
33. Roy S.C.; Funabiki N.; Munene, K.I.; Rahman M.M.; Kuribayashi M. An extension of active access-point configuration algorithm to IEEE 802.11n and 11ac dual interfaces in wireless local-area network. *Int. J. Future Comput. Commun.* **2022**, *11*, 18–26.
34. Raspberry Pi. Available online: <http://raspberrypi.org> (accessed on 24 November 2022).
35. Archer T4U. Available online: <https://www.tp-link.com/jp/home-networking/adapter/archer-t4u/> (accessed on 24 November 2022).
36. Funabiki N. *Wireless Mesh Networks*; InTech-Open: London, UK, 2011. Available online: <https://www.intechopen.com/books/26> (accessed on 24 November 2022).
37. Yan, D.; Zhang, C.; Liao, H.; Yang, L.; Li, P.; Yang, G. AP deployment research based on physical distance and channel isolation. *Abst. Appl. Anal.* **2014**, *2014*, 941547. [[CrossRef](#)]
38. Tang, S.; Ma, L.; Xu, Y. A novel AP placement algorithm based on user distribution for indoor WLAN system. *China Commun.* **2016**, *13*, 108–118. [[CrossRef](#)]
39. Tian, Y.; Huang, B.; Jia, B.; Zhao, L. Optimizing wifi ap placement for both localization and coverage. In Proceedings of the International Conference on Algorithms and Architectures for Parallel Processing, Guangzhou, China, 15–17 November 2018; pp. 489–503.
40. Wan, X.; Guan, X.; Shen, Y.; Choi, B.Y. Optimal user association in hybrid WLANs under bandwidth constraints. In Proceedings of the IEEE International Smart Cities Conference (ISC2), Kansas City, MO, USA, 16–19 September 2018; pp. 1–8.
41. Islam, M.T.; Choi, B.Y. Jointly maximizing throughput and utilization for dense enterprise WLANs. In Proceedings of the IEEE International Smart Cities Conference (ISC2), Casablanca, Morocco, 14–17 October 2019; pp. 753–759.
42. Zhi, Z.; Wu, J.; Meng, X.; Yao, M.; Hu, Q.; Tang, Z. AP deployment optimization in non-uniform service areas: A genetic algorithm approach. In Proceedings of the IEEE 90th Vehicular Technology Conference (VTC2019-Fall), Honolulu, Hawaii, USA, 22–25 September 2019; pp. 1–5.
43. Liu, P.; Meng, X.; Wu, J.; Yao, M.; Tang, Z. AP deployment optimization for WLAN: A fruit fly optimization approach. In Proceedings of the IEEE/CIC International Conference on Communications in China (ICCC), Changchun, China, 16–18 August 2018; pp. 478–483.
44. Liu, J.; Aoki, T.; Li, Z.; Pei, T.; Choi, Y.J.; Nguyen, K.; Sekiya, H. Throughput analysis of IEEE 802.11 WLANs with inter-network interference. *Appl. Sci.* **2020**, *10*, 2192. [[CrossRef](#)]
45. Kong, Z.; Wu, D.; Jin, X.; Cen, S.; Dong, F. Improved AP deployment optimization scheme based on multi-objective particle swarm optimization algorithm. *KSII Trans. Internet Infor. Syst. (TIIS)* **2021**, *15*, 1568–1589.
46. Qiu, S.; Chu, X.; Leung, Y.W.; Ng, J.K.Y. Joint access point placement and power-channel-resource-unit assignment for IEEE 802.11 ax-Based dense WiFi network with QoS requirements. *IEEE Trans. Mob. Comput.* **2021**. [[CrossRef](#)]

47. Abusubaih, M.A.; Eddin, S.N.; Khamayseh, A. IEEE 802.11n dual band access points for boosting the performance of heterogeneous WiFi networks. In Proceedings of the 8th ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks, New York, NY, USA, 3–8 November 2013; pp. 1–4.
48. Nishat, K.; Javed, F.; Salman, S.; Yaseen, N.; Fida, A.; Qazi, I.A. Slickfi: A service differentiation scheme for high-speed w lans using dual radio aps. In Proceedings of the 12th International on Conference on Emerging Networking EXperiments and Technologies (CoNEXT'16), New York, NY, USA, 12–15 December 2016; pp. 177–189.
49. Faria, D.B. Modeling Signal Attenuation in IEEE 802.11 Wireless LANs—Vol. 1. 2005. Available online: <http://www-cs-students.stanford.edu/~dbfaria/files/faria-TR-KP06-0118.pdf> (accessed on 24 November 2022).
50. Lu, T.; Funabiki, N.; Munene, K.I.; Sudibyoy, R.W. An improved throughput estimation model for concurrent communications of multiple hosts in wireless local-area network. In Proceedings of the IEEE Hiroshima Section Student Symposium (HISS), Okayama, Japan, 30 November–1 December 2019; pp. 360–363.
51. Wolsey, L.A. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica* **1982**, *2*, 385–393. [[CrossRef](#)]
52. Williamson, D.P.; Shmoys, D.B. *The Design of Approximation Algorithms*, 1st ed.; Cambridge University Press: Cambridge, UK, 2011.
53. Debnath, S.K.; Funabiki, N.; Lwin, K.S.; Al Mamun, M.S.; Sudibyoy, R.W.; Huda, S. Raspberry Pi configuration for access-point and its throughput measurements in IEEE802.11n wireless networks. *IEICE Tech. Rep.* **2016**, *116*, 101–106.
54. Hostapd. Available online: <https://w1.fi/hostapd/> (accessed on 24 November 2022).
55. Iperf—The TCP, UDP and SCTP Network Bandwidth Measurement Tool. Available online: <https://iperf.fr/> (accessed on 24 November 2022).
56. Islam, M.E.; Funabiki, N.; Lwin, K.S.; Mamun, M.S.A.; Debnath, S.K.; Kuribayashi, M. Performance evaluations of elastic WLAN testbed. *IEICE Gen. Conf.* **2016**, S76–S77.
57. Sudibyoy, R.W.; Funabiki, N.; Kuribayashi, M.; Munene, K.I.; Islam, M.M.; Kao, W.C. A TCP fairness control method for two-host concurrent communications in elastic WLAN system using Raspberry Pi access-point. In Proceedings of the 2nd International Conference on Communication Engineering and Technology (ICCET), Nagoya, Japan, 12–15 April 2019; pp. 76–80.
58. Islam, M.M.; Funabiki, N.; Kuribayashi, M.; Debnath, S.K.; Munene, K.I.; Lwin, K.S.; Sudibyoy, R.W. Dynamic access-point configuration approach for elastic wireless local-area network system and its implementation using Raspberry Pi. *Int. J. Netw. Comput.* **2018**, *8*, 254–281. [[CrossRef](#)]
59. Qualcomm. Band-Steering for Dual-Band Wi-Fi Access Points. Available online: https://www.qualcomm.com/content/dam/qcomm-martech/dm-assets/documents/qc_band_steering_wp.pdf (accessed on 8 February 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.