

Assigning Channels in WLANs with Channel Bonding: A Fair and Robust Strategy

Amel Chadda, Marija Stojanova, Thomas Begin, Anthony Busson,
Isabelle Guérin Lassous

Univ Lyon, UCB Lyon 1, ENS Lyon, Inria, CNRS, LIP UMR 5668 - Lyon, France

Abstract

The ever-growing popularity of WLANs has led to a numerous and highly diverse set of solutions for increasing the available link data rates. Amongst these solutions, channel bonding offers the possibility to use wider channels and increase the data rates by a factor of two, four, or eight. The issue of properly selecting the channels and their width remains a complex problem. In this paper, we present a fast, scalable, and fully graph-centric strategy for choosing a channel width and assignment for the APs of an IEEE 802.11-based WLAN. It typically outperforms strategies consisting of selecting the channel width regardless of the WLAN topology by 15% in fairness and 20% in throughput.

1. Introduction

The ever-growing popularity of WLANs (Wireless Local Area Networks) has led to a numerous and highly diverse set of solutions for increasing the available link data rates. Amongst these solutions, channel bonding was introduced in IEEE 802.11n allowing an Access Point (AP) to use two 20 MHz channels as a single 40 MHz channel. By using a bandwidth that is twice as large and limiting the number of used pilot channels in OFDM, the resulting data rates are slightly more than doubled as compared to those of a single 20 MHz channel. Channel bonding was further enhanced in IEEE 802.11ac/ax by offering the possibility to bond up to four or eight 20 MHz channels and forming a single 80 MHz or 160 MHz channel.

While the benefits of increased data rates are easily comprehensible, finely characterizing the drawbacks of channel bonding can be challenging. In a WLAN where channel access is decentralized, spatial reutilization is an important asset that ensures the coexistence of simultaneous transmissions within the same communication area. When APs use wider channels, a trade-off is made in favor of having higher data rates at the expense of lower number of simultaneous transmissions. It is often established that, in most cases, wider

Email addresses: amel.chadda@ens-lyon.fr (Amel Chadda),
marija.stojanova@ens-lyon.fr (Marija Stojanova), thomas.begin@ens-lyon.fr (Thomas Begin),
anthony.busson@ens-lyon.fr (Anthony Busson),
isabelle.guerin-lassous@univ-lyon1.fr (Isabelle Guérin Lassous)

channels favor the average throughput of APs while narrower channels may represent a better option when fairness among the APs is a priority. However, the issue of selecting the channels and their width remains a complex problem.

To the best of our knowledge, there is currently no simple recipe for selecting the appropriate channel width and assignment (a.k.a allocation). In this paper, we present a fast and robust strategy for choosing a channel width and assignment for the APs of an IEEE 802.11ax-based WLAN. The approach is fully graph-centric and highly tractable. The proposed solution is an extended and improved version of a work published in [4]: we introduce a larger set of randomly-generated graphs; we enhance the algorithm using more diverse training data; we extend our simulation results and deepen the analysis by varying the transmission parameters and the number of associated stations.

Overall, our solution helps restrain the number of APs experiencing severe starvation, hence leading to a good trade-off between the average throughput of APs and fairness. On average, it outperforms strategies consisting of selecting the channel width regardless of the WLAN topology by 15% in fairness and 20% in throughput. Additionally, the obtained results challenge the preconceived notion that wider channels generally result in higher average throughputs of APs and shed light upon the importance of the WLAN's topology and average degree when selecting a channel width.

The remainder of the paper is organized as follows. Section 2 provides an overview of the related work. In Section 3, we precisely describe the problem under study. Our proposed solution is presented in Section 4. We evaluate the precision and applicability its in Section 5 through a number of simulation experiments. Section 6 concludes this paper.

2. Related Work

In the first versions of IEEE 802.11, namely the initial standard and the amendments b/a/g, channels had a fixed width of 20 MHz. In this context, channel assignment (CA) algorithms consisted of assigning non-overlapping channels to APs with the aim of minimizing interference between APs [13, 10, 16, 19].

Since the IEEE 802.11n amendment, APs can aggregate channels using a mechanism called channel bonding. More precisely, the bonding can be applied to two 20 MHz channels in 802.11n, and two, four or eight 20 MHz channels for the most recent amendments (ac/ax) leading to channel widths of 40, 80, and 160 MHz, respectively. While wider channels provide higher physical data rates for transmitting frames, their use increases the number of overlapping channels, interference, and conflicts between APs. Overall, the introduction of channel bonding hardens the CA problem that typically aims at finding an appropriate trade-off between interference and throughput. In particular, the number of possible configurations is significantly increased.

The CA problem with channel bonding is referred to as CB (Channel Bonding) problem in the literature. We categorize solutions to the CB problem through three approaches: *i*) methods that consider a model to evaluate assignments and that try to find the assignment that optimizes a given objective function, *ii*) solutions that are based on measurements and adapt their configuration accordingly, and *iii*) algorithms that use machine learning techniques.

One of the pioneering works presenting a solution for the CB problem belongs to the first approach (model based). The solution named SA (Spectrum

Assignment for WLAN) is formulated as an optimization problem for which the authors proposed a distributed resolution [7]. For a given topology, the algorithm aims at minimizing interference between APs while taking into account the preferences of APs for certain channel widths. The authors of [12] present an analytical model that considers both collisions and interference. The CB problem applies to IEEE 802.11ac WLAN and is formulated to optimize the throughput given traffic demand. The optimization problem is then solved using a genetic algorithm. In [8] and [1], the model is based on a continuous time Markov chain (CTMC). The authors of [8] use this model to evaluate the throughput and the system utility for any WLAN topology. It is then combined into an optimization problem for which the authors proposed a heuristic. The authors of [1] use the CTMC on simple topologies to evaluate the performance of the WLAN as a function of the channel widths. Simulations are used for more complex scenarios. They show that, on average, an adaptation of the channel width on a per-packet basis outperforms single-channel transmissions. However, it may generate unfair situations where WLANs starve. A Markov network that models the interaction between the nodes is proposed in [2]. The model and simulations considering IEEE 802.11ac and ax amendments are applied to dense WLANs. Results show that spatial correlations between the nodes significantly impact the performance even between nodes that are not in the sensing range of each other. A renewal process is proposed in [11] to model IEEE 802.11ac and ax WLANs. Results on simple topologies allow the authors to derive the performance of channel bonding for these two amendments. Based on these results, a heuristic is proposed to select the channels.

The second category of algorithms uses a data-based approach where channels are set in real-time according to local measurements. In [14], a centralized solution is proposed to solve the CB problem. A controller collects the channel utilization data from the APs and updates a matrix whose elements represents the estimated utilization of a given channel when it is assigned to a particular AP. The channel assignment problem is solved by finding a solution that maximizes the sum of the channel utilization. Another online solution is proposed in [3]. As in [14], the algorithm is based on the activity of the channels. When an AP tests a new channel, it associates a satisfaction score based on what it has been able to send on this channel during a certain period. If the score is satisfactory, the AP remains on this channel, otherwise, it resumes its exploration. Two scenarios are studied in [23] corresponding to an enterprise and a residential WLAN. Their method to the CA problem considers the constraints imposed by bonding, and attempts to use wider bandwidths as they often result in higher throughput.

Machine learning (ML) techniques offer a promising approach to the CB problem. In [22], a neural network is employed and combined to a Markov chain. This allows the APs to predict the usage of different channels. The assignment is then performed to maintain fairness between channels and APs. Two algorithms based on reinforcement learning are proposed in [17] and [15]. They consist of a real-time exploration of new configurations and then exploiting the ones that offer good performance. A dense WLAN scenario is studied in [17]. The proposed method relies on a graph convolutional network to extract the carrier sensing relationships between APs. A game theory method is then used to collect the training set, on which a neural network is applied to perform the CA. The authors of [15] focus on a multi-armed bandits approach with the

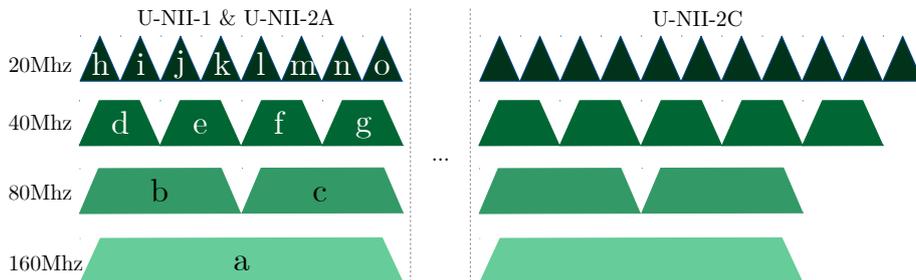


Figure 1: Available channels in the 5GHz band in Europe [6]. Only the channels of U-NII-1 and U-NII-2A are labeled as they are reused in Section 5.

Thompson sampling algorithm to select new configurations to evaluate. In their work, the solution performs both the CA and the associations between APs and stations.

In summary, solutions based either on models or machine learning techniques must explore a tremendous number of assignments and channel widths. They typically improve the current assignment in terms of fairness or throughput, but they often involve a very limited exploration of the solution space. As for on-line and ML techniques, their exploration phase incurs a cost as new configurations that can be poor in terms of performance have to be tested continuously. In this paper, we split the solution space into its two dimensions, i.e., channel and width. By doing so, we significantly reduce the complexity of the CA problem at the cost of having a less fine-grained exploration. Note that the exploration of solutions is performed using an analytical model so that only the solution chosen as most appropriate is applied to the WLAN. This can prove to be a useful property when trying to avoid disrupting an operational network.

3. Terminology

Before presenting our proposed strategy to select channel width and assignment, we introduce some terminology used to describe WLANs and performance metrics of interest for our study.

3.1. Physical and logical neighbors

We consider an IEEE 802.11 WLAN composed of N APs that use Static Channel Bonding (SCB) [9]. The APs n and m ($\in \{1, \dots, N\}^2$) of the WLAN are said to be *physical neighbors* if they can detect each other's transmissions. Note that the physical neighbors of an AP are mainly determined by the transmission power and sensitivity threshold of its NIC (Network Interface Card) and by the radio wave propagation. We assume that all detection ranges are symmetrical, i.e., if n is a physical neighbor of m , then m is also a physical neighbor of n . A sample four-AP WLAN is shown in Figure 2a, where AP 3 detects all the other APs, while AP 4 detects only the transmissions of AP 3. Thus AP 3 is the only physical neighbor of AP 4 as depicted in Figure 2b.

Each AP operates either on a basic channel of 20 MHz, or on a bonded channel of 40 MHz, 80 MHz, or 160 MHz. The number of non-overlapping channels for a given width will depend on the chosen width, but also on regional

regulations. For example, Figure 1 shows the available channels in Europe in the 5 GHz band. In our solution, we use the same channel width for all APs, and we use w to denote its value and C_w to denote the set of available channels for the width w MHz. For instance, for $w = 160$ MHz in Europe and in the U-NII-1 and U-NII-2A bands, we have $C_w = \{a\}$ and $|C_w| = 1$, as shown in Figure 1. It is important to mention that in real-life WLANs, it may happen that two APs that detect each other’s transmissions on a 20 MHz channel no longer do so on a wider channel, as the energy is spread over a wider spectrum.

For the sake of our study, we introduce the notion of *logical neighbors*. Logical AP neighbors are APs that are both physical neighbors and operating on overlapping channels. Clearly, unless there is only one channel in use for the whole WLAN, there are typically much less logical neighbors than physical neighbors thanks to the use of non-overlapping channels.

3.2. Physical and logical conflict graphs

We associate two graphs to any WLAN: its *physical conflict graph*, and its *logical conflict graph*. The former derives only from the pairs of physical neighbors of each AP. Its vertices represent the WLAN’s APs and an edge exists between two vertices when the corresponding APs are physical neighbors. Figure 2b depicts the physical conflict graph associated to the WLAN shown in Figure 2a.

The logical conflict graph depends on the chosen channel assignment and hence on the logical neighbors. We use $\mathcal{G}_w(v)$ to denote the logical conflict graph associated to a channel of width w and an assignment v where v is a vector describing the channels assigned to each AP. For instance, $v = (b, b, c, b)$ represents a possible channel assignment for our sample four-AP WLAN when the channel assignment is as follows: APs 1, 2, and 4 are all using the same channel of 80 MHz, whereas AP 3 is using another channel of 80 MHz (depicted by the dashed line). Note that, with this channel assignment, AP 3 is no longer in conflict with any of its neighbors and the logical conflict graph includes only one edge between APs 1 and 2. The corresponding logical conflict graph depicted in Figure 2c is denoted by $\mathcal{G}_{80}(b, b, c, b)$.

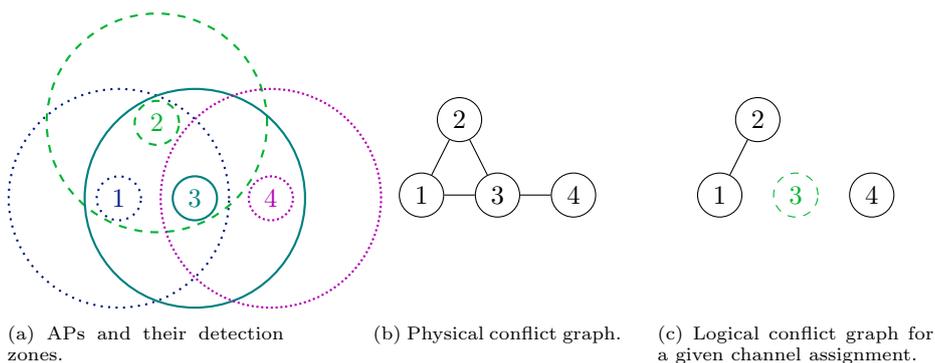


Figure 2: An example of a WLAN with 4 APs.

3.3. Performance metrics

Each AP of the WLAN exchanges traffic with its associated station(s). The traffic is characterized by the frame’s payload length L , the HE-MCS index

Notation	Description
N	Number of APs in the WLAN
$\mathcal{G}_w(v)$	Logical conflict graph for channel width w and channel assignment described by vector v
C_w	Set of available channels of width w
L	IP payload length, in bytes
R	Physical data rate in Mbps for a given MCS index, number of spatial streams and guard interval length
a	A-MPDU rate, $a \in \{1, \dots, 64\}$ MPDUs
A_n	Achievable throughput of AP n in Mbps
B_n	Achieved throughput of AP n in Mbps
τ	Starvation threshold in Mbps
TH	Average throughput of APs for the WLAN in Mbps
ST	Starvation index (number of APs in starvation)
PF	Proportional fairness in Mbps

Table 1: Principal notation.

(High-Efficiency Modulation and Coding Scheme) of 802.11ax, the number of spatial streams, the guard interval length, and the A-MPDU rate a (i.e., the maximum number of MPDUs aggregated in every frame transmission). From these parameters, we can derive the *achievable throughput* of AP n , denoted by A_n , as:

$$A_n = \frac{L \times a}{T_{DCF} + a \times \frac{L + H_{MAC}}{R} + T_{ACK}}, \quad (1)$$

where T_{DCF} is the sum of the overhead times needed for the DCF procedure [6] including the physical layer header, H_{MAC} is the MAC header in bytes, R is the physical data rate resulting from the combination of MCS index and guard interval length, and T_{ACK} is the total time needed to send the acknowledgment frame. The achievable throughput A_n is simply the throughput AP n would achieve in downlink if it were the only AP to access its channel. However, this is often not the case and the AP has to share the channel with other APs (i.e., its logical neighbors) resulting in a so-called *achieved throughput*. We denote AP n 's achieved throughput as B_n , and it follows that $B_n \leq A_n$. The average throughput of APs of the WLAN, denoted TH , is simply computed as the sum of the achieved throughputs of all the APs divided by the number of APs:

$$TH = \frac{\sum_{n=1}^N B_n}{N}. \quad (2)$$

Defining starvation is a bit more problematic as it depends on both the interpretation and the performance goals. In this paper, we consider that an AP is experiencing starvation whenever its achieved throughput B_n is lower than some fixed threshold denoted by τ . We use ST to denote the number of APs in starvation:

$$ST = \sum_{n=1}^N \mathbb{1}_{B_n < \tau}. \quad (3)$$

Furthermore, for the sake of completeness, we also consider the *proportional fairness* metric that we compute here as follows:

$$PF = \left(\prod_{n=1}^N B_n \right)^{1/N}. \quad (4)$$

Note that PF is a non-negative real number expressed in Mbps.

Having defined our WLAN representation and the performance metrics of interest for our study, we now detail our proposed solution for radio channel assignment.

4. Proposed Solution

We now describe our solution for channel selection that circumvents the original problem's complexity by decoupling the channel width and the channel assignment issues.

4.1. Maximum Independent Set Ratio

Our proposed solution is based on the WLAN's conflict graphs, i.e., its physical conflict graph and the logical conflict graphs arising from different channel assignments. It relies on the identification of the APs that are in advantageous positions regarding transmission opportunities, or equivalently, those most likely to be in starvation because of their location in the conflict graph.

We first recall some graph-theoretic definitions useful for our study. The *maximal independent set* refers to a subset of vertices that contains no neighbor nodes and cannot be extended by adding more vertices. For example, in our sample four-AP conflict graph of Figure 2b, there are three maximal independent sets: $\{1, 4\}$, $\{2, 4\}$, and $\{3\}$. A *maximum independent set* denotes the maximal independent set(s) with the highest cardinality. In our example, $\{1, 4\}$ and $\{2, 4\}$ are both maximum independent sets. We use $MMS(\mathcal{G}_w(v))$ to denote the set containing all the maximum independent sets of $\mathcal{G}_w(v)$.

We can now introduce a new quantity that we specifically devise for our study: the *Maximum Independent set Ratio (MIR)*. MIR is calculated for each of the graph's vertices as the proportion of maximum independent sets to which the vertex belongs. Therefore, for vertex n , we have:

$$MIR(n, \mathcal{G}_w(v)) = \frac{|K|}{|MMS(\mathcal{G}_w(v))|}. \quad (5)$$

where K is the subset of $MMS(\mathcal{G}_w(v))$ that verifies $n \in K$.

The maximum independent sets and the MIR values of any graph and any vertex can be obtained using the Bron-Kerbosch algorithm [21]. We discuss the complexity of this algorithm in Section 4.3.

As an example, the MIR values for the vertices representing our four-AP sample network when using a single channel of 160 MHz for all APs (i.e., $v = (a, a, a, a)$) are shown in Figure 3. AP 4 belongs to all maximum independent sets and so it has an $MIR(4, \mathcal{G}_{160}(a, a, a, a)) = 1$. APs 1 and 2 belong to half of the maximum independent sets, so we have $MIR(1, \mathcal{G}_{160}(a, a, a, a)) = MIR(2, \mathcal{G}_{160}(a, a, a, a)) = 0.5$. Finally, we have $MIR(3, \mathcal{G}_{160}(a, a, a, a)) = 0$ since AP 3 does not belong to any maximum independent set.

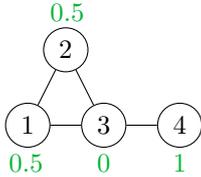


Figure 3: Calculated *MIRs* (Maximum Independent set Ratio) for a given logical conflict graph.

Interestingly, the *MIR* of vertex n is a major factor in determining its achieved throughput B_n . Let us consider a (training) set of 52 randomly generated physical conflict graphs of WLANs containing between 10 and 25 APs and with average AP degrees ranging from 2.9 to 8.5. Recall that the degree of a vertex (an AP in our case) is the number of edges containing it, and that the average degree of a graph is simply computed as twice the number of edges, divided by the number of vertices. We assume that the APs are in saturation and sending downlink traffic to their associated station using MCS (Modulation and Coding Scheme) index 5, a frame aggregation rate $a = 4$ MPDUs, a payload $L = 1400$ bytes, and a guard interval of 3200 ns. We use the ns-3 discrete-event network simulator [18] to evaluate the achieved throughput B_n of each AP. Figure 4 depicts the AP's achieved throughput as a function of its *MIR* for every AP of these graphs when two channel widths are used: 20 and 160 MHz. We observe that, in either case, there is a strong correlation between an AP's *MIR* and its achieved throughput B_n . Note that similar correlations (not shown in this paper) hold for channel widths of 40 and 80 MHz.

This observation is the cornerstone of our proposed solution: for a given logical conflict graph, we can easily calculate the *MIR* value of each AP that, in turn, helps us estimate its attained throughput and hence predict whether that AP is in starvation or not. To do that, we naturally resort to a linear regression as follows:

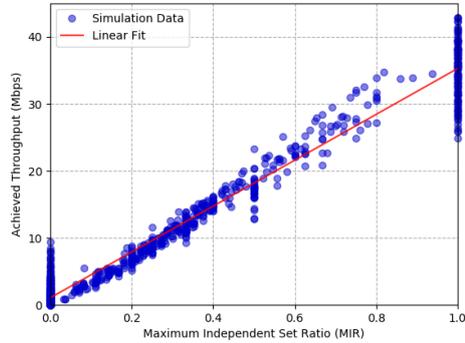
$$B = \beta_{0,w} + \beta_{1,w} \times \text{MIR} , \quad (6)$$

with $\beta_{0,w}$ and $\beta_{1,w}$ representing the linear coefficient found for a channel width of w MHz.

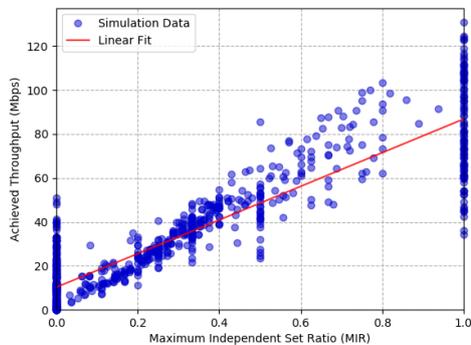
The intuition behind the *MIR* lies in the fact that the mechanism of medium access in WLANs tends to maximize the number of simultaneous transmissions [5]. Therefore, APs that belong to many maximum independent sets (and have high *MIR* values) are more likely to be in transmission and achieve higher throughputs. Figure 4 shows that there are two points where estimating throughput based on *MIR* is not as precise, namely the extremes $\text{MIR} = 0$ and $\text{MIR} = 1$. However, the simulation results of Section 5 show that we can use Eq. (6) to estimate the achieved throughput even for these two extreme values of *MIR* with a satisfactory accuracy.

4.2. Channel width and assignment algorithm

In our solution, we perform the search for the channel assignment in the worst-case scenario, namely when all APs are saturated (permanently needing to access the channel). As it was shown in [1], we rely on the assumption that typically wider channels favor the average throughput of APs of the WLAN,



(a) Channel width: 20 MHz.



(b) Channel width: 160 MHz.

Figure 4: The achieved throughput as a function of MIR (Maximum Independent set Ratio) obtained by simulating the behavior of hundreds of APs for 52 different WLANs.

while narrower and separate channels avoid starvation. Hence, our solution iterates on all the possible channel widths, starting with the widest channel of 160 MHz. For each channel width, we use the Tabu Search algorithm [20] to compute a k -coloring of the physical conflict graph, where k is the number of available channels for the considered channel width. In a nutshell, the Tabu search attempts to find a channel assignment that minimizes the number of edges in the resulting logical conflict graph. We then calculate the MIR of each AP and estimate its achieved throughput using the linear regression model. At this stage, we are able to compute ST , i.e., the number of APs in starvation. If ST is larger than zero, we divide the channel width in half and multiply the number of available channels by two. With this new setting, we run the same procedure: coloring the graph, calculating the APs' MIR s and then the ST metric, and if there are starving nodes we divide the channel width once again. The procedure ends when either a channel width for which there are no starving APs is found, or when the minimum channel width of 20 MHz is reached. The pseudo-algorithm of the complete solution is given in Algorithm 1.

Note that Algorithm 1 has a single tuning parameter: the starvation threshold, τ . If its value is too large, this could cause the algorithm to erroneously

Algorithm 1 Solution to select channel width and assignment

```
1: Input: physical conflict graph, number of vertices  $N$ , starvation threshold  $\tau$ 

2: Output: selected channel width  $w$ , channel assignment  $v$ 



---


3:  $w \leftarrow 160$ 
4:  $k \leftarrow |C_{160}|$  // # of channels for the current width
5: while  $w \geq 20$  do
6:    $v \leftarrow$  compute a  $k$ -coloring channel assignment (Tabu)
7:   compute  $MIR(n, \mathcal{G}_w(v)), \forall n \in \{1, \dots, N\}$  using Eq. 5
8:   compute  $B(n), \forall n \in \{1, \dots, N\}$  using Eq. 6
9:   compute  $ST$  using Eq. 3
10:  if  $ST > 0$  then
11:     $w \leftarrow w / 2$ 
12:     $k \leftarrow |C_w|$ 
13:  else
14:    return  $(w, v)$ 
15:  end if
16: end while
17: return  $(w, v)$ 



---


```

suppose APs in starvation and to wrongfully reject an otherwise appropriate channel assignment. Conversely, a too small value for τ can lead the algorithm to overlook APs in starvation and lead to channel assignment causing starvation.

In the interest of clarity, Figure 5 shows the results returned by Algorithm 1 on the sample four-AP WLAN using a starvation threshold $\tau = 5$ Mbps. For the sake of this example, we assume that there is only a single 160 MHz channel so that $|C_{160}| = 1$. The main round of the algorithm was executed twice, namely for a channel width of 160 MHz and then 80 MHz. In the first round, we have $MIR(3, \mathcal{G}_{160}(a, a, a, a)) = 0$, AP 3's estimated attained throughput is then $B_3 = 0$, i.e., AP 3 is in starvation. Thus, we have $ST > 0$ and the algorithm rejects the channel width of 160 MHz. The algorithm now considers a channel width of 80 MHz. For this second round, we obtain $MIR(3, \mathcal{G}_{80}(b, b, c, b)) = 1$ leading to an estimated attained throughput of 20 Mbps that is above the starvation threshold. Because APs 1, 2, and 4 have MIR s of 0.5, 0.5, and 1, respectively, our algorithm estimates that they are also above the starvation threshold and therefore all APs have a satisfactory throughput. As a result, the algorithm ends and returns $w = 80$ MHz and $v = (b, b, c, b)$ for the channel assignment as its solution.

4.3. Complexity analysis

The complexity of the proposed solution is highly dependent on the complexity of the underlying algorithms. The Tabu Search algorithm that we use to find a k -coloring has an $O(D_{max}N^4)$ complexity (when the objective function is to minimize the number of edges in the logical conflict graph), where D_{max} is the maximum degree in the graph [20]. Bron-Kerbosch's algorithm that is used to find the MIR ratios has an $O(3^{\frac{N}{3}})$ complexity [21]. All other parts of our

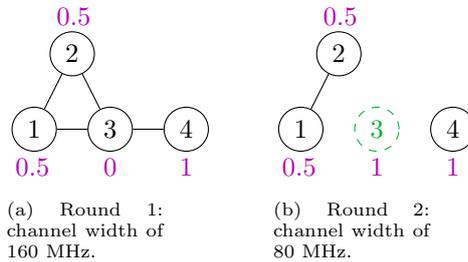


Figure 5: Logical conflict graph and *MIR* values computed by Algorithm 1 on the example of the four-AP WLAN of Figure 2b.

	Very sparse	Sparse	Dense	Very dense	Ultra dense
Average degree, d	$d < 5$	$5 \leq d < 7$	$7 \leq d < 9$	$9 \leq d < 11$	$11 \leq d$
Numbers of graphs	19	22	18	14	15

Table 2: Graph distribution by average degree.

algorithm have a lower complexity, meaning that the overall complexity of the proposed solution is $O(3^N)$.

It should be noted that the use of the Tabu Search and the Bron-Kerbosch algorithms is fully independent of the rest of the solution. Moreover, other objective functions than the minimization of the number of edges in the logical conflict graph can be used as well. These possible modifications, as well as the choice of starvation threshold, create an easily adaptable framework suited for different WLAN performance goals.

5. Simulation Results

For the sake of our simulation study, we consider a large set of 88 WLANs that together form our validation set. Note that these WLANs differ from those used previously to train our solution in Section 4, i.e., learn the parameters of the linear regression estimating TH as a function of MIR . In this validation set, WLANs comprise between 8 and 30 APs. The physical conflict graphs of these 88 WLANs, which derive from the spatial arrangement of their APs, were randomly generated and they exhibit average degrees ranging from 2.5 to 15. As shown in Table 2, we label the WLANs as being very sparse, sparse, dense, very dense, or ultra dense depending on their average degree.

For each AP of the WLANs, we let the number of associated stations be uniformly distributed in the range [2,5]. The APs are sending downlink traffic to their associated stations using a frame aggregation rate $a = 4$ MPDUs with a payload of 1400 bytes and a uniform distribution of MCS indexes 0 to 11, and a guard interval of 3200 ns. We study each WLAN in saturation conditions, i.e., the transmission queue at each of their AP is always nonempty. By considering this worst-case scenario, we investigate the behavior of our solution when pushed to extreme conditions. Note that if our solution succeeds in the worst case, then it can only do better in more favorable conditions.

The set of available channels contains all the channels from the U-NII-1 and U-NII-2A bands accessible in Europe (see Figure 1), i.e., one channel of 160 MHz, two of 80 MHz, four of 40 MHz, and eight channels of 20 MHz.

We consider the three metrics of interest as defined in Section 3.3:

- *TH*: the average throughput of APs in Mbps (see Eq. (2)),
- *ST*: the number of starving APs (see Eq. (3)),
- *PF*: the proportional fairness in Mbps (see Eq. (4)).

Throughout this section, we evaluate the performance gain of our solution as opposed to baseline solutions under different configurations. The baseline configurations include *i*) a single 160 MHz channel (i.e., the physical and logical conflict graphs always coincide) and *ii*) eight appropriately-assigned channels of 20 MHz (i.e., almost no conflicts in the logical conflict graph). As for the changing parameters, we test different values of the starvation threshold τ , evaluate the impact of average degrees, and assess the robustness of our approach when the traffic parameters are varying.

We implemented our solution in C++ and we then use ns-3 and the IEEE 802.11ax standard amendment to evaluate the channel assignments proposed by our solution and by other common strategies. Table 3 reports the simulation settings.

Parameter	Value
ns-3 version	3.32
Simulation duration	3 seconds
Frequency band	5 GHz
MCS Control	HeMcs (High Efficiency)
IP Payload length, L	1400 bytes
Transport Protocol	UDP
Maximum A-MPDU length	4 IP packets
Number of APs	8 - 30
Number of stations per APs	2 - 5

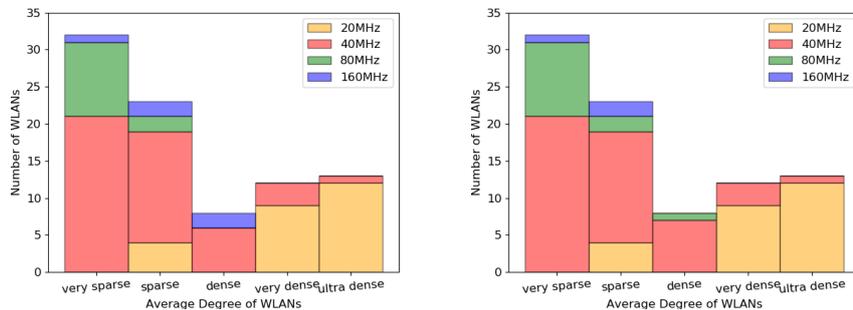
Table 3: Simulation settings.

5.1. Starvation threshold and average degree

We begin by examining how the choice of τ , the starvation threshold, impacts the channel width chosen by our solution for WLANs with different average degrees. Figure 6 shows the results for $\tau = 1$ Mbps (Figure 6a) and $\tau = 5$ Mbps (Figure 6b). We first notice that 40 MHz is the most commonly chosen channel width, in around 50% of the cases. As expected, wider channels are more frequently chosen when the average degree is fairly low, and a channel of 20 MHz is never proposed when the network is very sparse. On the other side of the spectrum, when the network is very or ultra dense, the only chosen channel widths are those of 20 MHz and 40 MHz. Figure 6 also suggests that the impact of the starvation threshold τ is fairly minimal. The distribution of recommended channel widths remains roughly the same when τ is increased from 1 to 5 Mbps.

5.2. Comparison with baseline solutions

Properly configuring channel width is highly dependent on the performance metric of interest. It is well accepted that, in general, narrower channels favor fairness, and wider channels provide higher average throughput of APs [1].



(a) For a starvation threshold of $\tau = 1$ Mbps. (b) For a starvation threshold of $\tau = 5$ Mbps.

Figure 6: Distribution of the channel widths selected by our solution as a function of the average degree of WLANs.

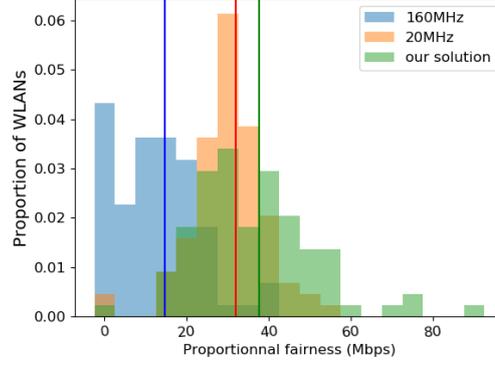
This reasoning leads to two opposing, but equally relevant, strategies: *i*) using multiple narrow, but independent, channels, or *ii*) using less, but wider, channels.

We use these two strategies as baselines when studying the performance of our proposed solution. The first strategy consists in assigning all APs to the 160 MHz channel (channel *a* in Figure 1), with the aim of maximizing average throughput of APs. The second strategy is assigning as many independent channels as possible, or the eight 20 MHz channels, using an appropriate channel allocation. In our case, we use the Tabu Search algorithm to find the allocation that minimizes the number of conflicts in the logical conflict graph. Finally, we compare the obtained *PF*, *ST*, and *TH* metrics for these two strategies and our proposed solution.

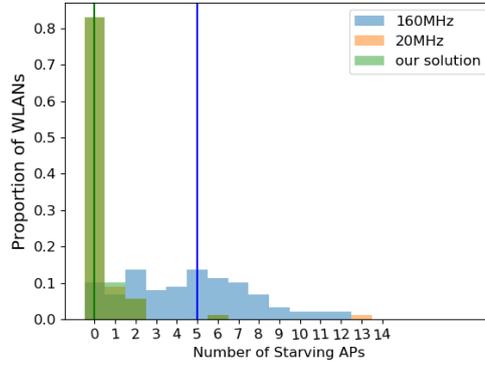
Figure 7 shows the corresponding results for $\tau = 5$ Mbps. Note that, in the interest of brevity, we show the results for a single τ value and that similar results were obtained for larger and smaller τ thresholds. The vertical lines in the figure show the median values found by the three strategies labeled "160 MHz", "20 MHz", and "Our solution". For all three metrics, the 160 MHz channel solution performs worse than the two other strategies. It is the only strategy that experiences some starvation (a median of 5 APs in starvation), which is not compensated by a higher average throughput of APs. Results show that our solution leads to both a higher *PF* and a higher *TH* than the eight 20 MHz channel strategies.

5.3. Varying traffic parameters: MCS index

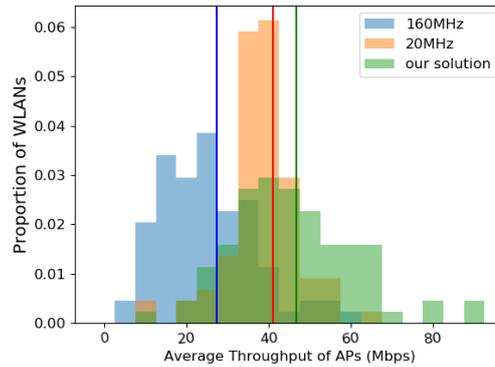
We now test how well our solution performs under more specific traffic conditions. We are interested in how the MCS indexes, or their distribution, can affect the performance of our proposed solution. Using the same two baseline strategies as before, we compare the obtained *PF*, *ST*, and *TH* values for different ranges of MCS indexes on the same collection of 88 graphs. We define four such ranges: low MCS (from 0 to 3), medium MCS (from 4 to 7), high MCS (from 8 to 11), and all MCS (from 0 to 11). Then, the MCS index of each pair of AP and station is uniformly chosen in the given range. As before, the number of associated stations per AP is uniformly distributed in the interval [2, 5].



(a) *PF*: Proportional Fairness.



(b) *ST*: Number of Starving APs.



(c) *TH*: Average throughput of APs.

Figure 7: Distribution of the performance metrics for the 88 WLANs of the validation set using 3 different strategies with a starving threshold $\tau = 5$ Mbps.

Table 4 summarizes the results. We observe that, in the considered scenarios, both our solution and using 20 MHz channels always offer a configuration with no starving nodes (i.e., $ST = 0$). Using a channel of 160 MHz can cause an

	<i>PF</i> (Mbps)			<i>ST</i> (nb of APs)			<i>TH</i> (Mbps)		
	160 MHz	20 MHz	Our sol.	160 MHz	20 MHz	Our sol.	160 MHz	20 MHz	Our sol.
MCS 0-3	7.70	13.54	17.69	7	0	0	17.54	15.16	21.18
MCS 4-7	19.22	43.60	49.74	3	0	0	32.11	44.17	54.10
MCS 8-11	23.68	65.97	69.25	2	0	0	36.38	66.20	75.35
All MCS	14.65	31.99	37.72	5	0	0	27.36	41.02	46.79

Table 4: Comparing the performance of the 3 strategies (160 MHz, 20 MHz, Our solution) in terms of *PF* (Proportional fairness), *ST* (number of starving APs), and *TH* (Average throughput of APs) for different MCS configurations and 88 WLANs. We use the following color code: Green designates the best strategy with regards to the performance metric in the multi-column; Red designates the worst strategy; Yellow designates the in-between strategy.

average as high as 7 starving APs in the case of MCS ranging from 0 to 3. Our solution consistently offers the highest *TH* and *PF* median values. However, the degree of improvement of these metrics is highly varying. Table 5 shows the average improvement provided by our solution as opposed to the two other strategies. On average, we improve the *PF* by 16.9% (resp. 159.60%) and the *TH* by 22.53% (resp. 66.84%) when compared to using eight 20 MHz channels (resp. a single 160 MHz channel).

	<i>PF</i> 160 MHz	<i>PF</i> 20 MHz	<i>TH</i> 160 MHz	<i>TH</i> 20 MHz
MCS 0-3	129.70%	30.65%	20.75%	39.71%
MCS 4-7	158.79%	14.08%	68.48%	22.48%
MCS 8-11	192.44%	4.97%	107.12%	13.82%
All MCS	157.47%	17.91%	71.02%	14.07%
Average	159.60%	16.90%	66.84%	22.52%

Table 5: Relative increase for *PF* (Proportional fairness) and *TH* (Average throughput of APs) observed over 88 WLANs when using our solution in place of each baseline strategy (viz. a single channel of 160 MHz or 8 channels of 20 MHz each).

6. Conclusions and Future work

In this paper, we presented a fast and robust solution for channel width selection and channel assignment in 802.11 WLANs using channel bonding. The proposed solution chooses a single channel width for all the APs of the WLAN that aims to avoid starvation in any of the network’s APs.

By introducing a novel, graph-centric metric, and by decoupling the channel width selection and the channel assignment, we offer a scalable approach that bypasses the usual complexity issues of classic channel assignment schemes. The approach is also fully adaptable to different performance goals, as it is a collection of several algorithms that can be modified separately and independently.

Simulation results using the ns-3 discrete-event network simulator and the most recent IEEE 802.11ax standard amendment show that the proposed solution is robust to different traffic parameters and network densities. It typically outperforms strategies consisting of selecting the channel width regardless of the WLAN topology by 15% in fairness and 20% in throughput.

As future work on the choice of radio channel width, we will seek to go beyond the objective of closing fairness gaps among the APs. We will investigate if we can adapt our solution to improve fairness among the stations of the WLAN.

References

- [1] S. Barrachina-Muñoz, F. Wilhelmi, and B. Bellalta. 2019. To overlap or not to overlap: Enabling channel bonding in high-density WLANs. *Computer Networks* 152 (2019), 40 – 53. <https://doi.org/10.1016/j.comnet.2019.01.018>
- [2] S. Barrachina-Muñoz, F. Wilhelmi, and B. Bellalta. 2020. Dynamic Channel Bonding in Spatially Distributed High-Density WLANs. *IEEE Transactions on Mobile Computing* 19, 4 (2020), 821–835.
- [3] S. Barrachina-Muñoz, F. Wilhelmi, and B. Bellalta. 2020. Online Primary Channel Selection for Dynamic Channel Bonding in High-Density WLANs. *IEEE Wireless Communications Letters* 9, 2 (2020), 258–262.
- [4] Amel Chadda, Marija Stojanova, Thomas Begin, Anthony Busson, and Isabelle Guérin Lassous. 2020. Towards a fast and efficient strategy to assign channels in WLANs with channel bonding. In *17th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks (PE-WASUN)*. 93–100.
- [5] Mathilde Durvy, Olivier Dousse, and Patrick Thiran. 2009. Self-organization properties of CSMA/CA systems and their consequences on fairness. *IEEE Transactions on Information Theory* (2009).
- [6] M. S. Gast. 2013. *802.11 ac: a survival guide: Wi-Fi at gigabit and beyond*. O’Reilly Media, Inc.
- [7] J. Herzen, R. Merz, and P. Thiran. 2013. Distributed spectrum assignment for home WLANs. In *32nd IEEE International Conference on Computer Communications (INFOCOM)*. 1573–1581.
- [8] C. Kai, Y. Liang, X. Hu, Z. Liu, and L. Wang. 2019. An effective channel allocation algorithm to maximize system utility in heterogeneous DCB WLANs. *Computer Networks* 153 (2019), 23 – 35. <https://doi.org/10.1016/j.comnet.2019.01.043>
- [9] Caihong Kai, Yuting Liang, Tianyu Huang, and Xu Chen. 2017. To bond or not to bond: An optimal channel allocation algorithm for flexible dynamic channel bonding in WLANs. In *86th IEEE Vehicular Technology Conference (VTC2017-Fall)*. IEEE.
- [10] B. Kauffmann, F. Baccelli, A. Chaintreau, V. Mhatre, K. Papagiannaki, and C. Diot. 2007. Measurement-Based Self Organization of Interfering 802.11 Wireless Access Networks. In *26th IEEE International Conference on Computer Communications (INFOCOM)*.
- [11] S. Khairy, M. Han, L. X. Cai, Y. Cheng, and Z. Han. 2019. A Renewal Theory Based Analytical Model for Multi-Channel Random Access in IEEE 802.11ac/ax. *IEEE Transactions on Mobile Computing* 18, 5 (2019), 1000–1013.
- [12] S. Lee, T. Kim, S. Lee, K. Kim, Y. H. Kim, and N. Golmie. 2019. Dynamic Channel Bonding Algorithm for Densely Deployed 802.11ac Networks. *IEEE Transactions on Communications* 67, 12 (2019), 8517–8531.

- [13] D.J. Leith, P. Clifford, V. Badarla, and D. Malone. 2012. WLAN channel selection without communication. *Computer Networks* (2012). <https://doi.org/10.1016/j.comnet.2011.12.015>
- [14] T. H. Lim, W. S. Jeon, and D. G. Jeong. 2016. Centralized channel allocation scheme in densely deployed 802.11 wireless LANs. In *18th International Conference on Advanced Communication Technology (ICACT)*. 249–253.
- [15] A. López-Raventós and B. Bellalta. 2020. Concurrent decentralized channel allocation and access point selection using multi-armed bandits in multi BSS WLANs. *Computer Networks* 180 (2020), 107381. <https://doi.org/10.1016/j.comnet.2020.107381>
- [16] A. Mishra, V. Brik, S. Banerjee, A. Srinivasan, and W. Arbaugh. 2006. A Client-Driven Approach for Channel Management in Wireless LANs. In *25th IEEE International Conference on Computer Communications (INFOCOM)*.
- [17] K. Nakashima, S. Kamiya, K. Ohtsu, K. Yamamoto, T. Nishio, and M. Morikura. 2019. Deep Reinforcement Learning-Based Channel Allocation for Wireless LANs with Graph Convolutional Networks. In *90th IEEE Vehicular Technology Conference (VTC2019-Fall)*. 1–5.
- [18] ns3 2021. The Network Simulator ns-3. <https://www.nsnam.org/>.
- [19] E. Rozner, Y. Mehta, A. Akella, and L. Qiu. 2007. Traffic-Aware Channel Assignment in Enterprise Wireless LANs. In *IEEE International Conference on Network Protocols (ICNP)*.
- [20] A. P. Subramanian, H. Gupta, S. R. Das, and J. Cao. 2008. Minimum Interference Channel Assignment in Multiradio Wireless Mesh Networks. *IEEE Trans. Mob. Comput.* (2008). <https://doi.org/10.1109/TMC.2008.70>
- [21] E. Tomita, A. Tanaka, and H. Takahashi. 2004. The Worst-Case Time Complexity for Generating All Maximal Cliques. In *10th Annual International Conference Computing and Combinatorics (COCOON)*. Springer. https://doi.org/10.1007/978-3-540-27798-9_19
- [22] J. Webber, A. Mehbodniya, K. Yano, and Y. Suzuki. 2019. Optimized WLAN Channel Allocation based on Gibbs Sampling with Busy Prediction using a Probabilistic Neural Network. In *International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*. 1–5.
- [23] A. Zakrzewska and L. Ho. 2019. Dynamic Channel Bandwidth Use through Efficient Channel Assignment in IEEE 802.11ac Networks. In *90th IEEE Vehicular Technology Conference (VTC2019-Fall)*.