

A Network Resource Management Framework for Wireless Mesh Networks

Jinqiang Yu¹  · Wai-Choong Wong²

Published online: 2 February 2017
© Springer Science+Business Media New York 2017

Abstract In infrastructure wireless mesh networks (WMNs), a mobile station (STA) associates with one of the nearby mesh access points (MAPs) that are connected to a capacity-limited wireless multi-hop backhaul. For better network resource utilization in WMNs, it is preferred that more traffic is carried by the good-backhaul MAPs that are closer to the portal or having better backhaul condition. In this paper, by taking the features of WMNs and the inter-cell interference into consideration, we propose a network resource management framework that jointly manages MAP–STA association, MAP access network channel assignment, and user bandwidth allocation. The proposed framework is composed of three components: an optimization-based association control algorithm that improves the network resource utilization by associating more STAs with the good-backhaul MAPs, an access network channel assignment algorithm that effectively increases the network capacity by reducing the interference at the good-backhaul MAPs, and a utility-fair bandwidth allocation algorithm that is flexible in adjusting the trade-off between network throughput and user fairness. We demonstrate the superior performance of the proposed algorithms through simulations with various network topologies and conditions.

Keywords Association control · Bandwidth allocation · Channel assignment · Wireless mesh networks

This work was carried out at the SeSaMe Centre. It was supported by the Singapore NRF under its IRC@SG Funding Initiative and administered by the IDMPO.

✉ Jinqiang Yu
yujinqiang@u.nus.edu

Wai-Choong Wong
elewwcl@nus.edu.sg

¹ Institute for Infocomm Research, Agency for Science, Technology and Research, 1 Fusionopolis Way, 21-01 Connexis South Tower, Singapore 138632, Singapore

² Department of Electrical and Computer Engineering, National University of Singapore, 4 Engineering Drive 3, Singapore 117576, Singapore

1 Introduction

In a wireless mesh network (WMN), mesh access points (MAPs) are connected to a wireless multi-hop backhaul that carries the data traffic of mobile stations (STAs) [1]. The wireless backhaul enables flexible and economic network deployment at the expense of lower network capacity compared to the wired backhaul in the conventional wireless local area networks (WLANs) [2]. To make better use of the network resource in WMNs, both the access network condition and the backhaul network condition should be taken into consideration when allocating network resources such as radio frequencies and transmission opportunities. Currently most of the resource management schemes designed for WLANs consider the access network only, assuming an abundant backhaul capacity, which is not the case in WMNs, and therefore cannot be directly applied in WMNs. In this paper, we propose a resource management framework for WMNs that takes the features of WMNs into consideration and achieves better network resource utilization by jointly managing MAP channel assignment, MAP–STA association, and user bandwidth allocation.

We consider WMNs that consist of three types of nodes: STA, MAP and portal. A STA must associate with one of the MAPs in its vicinity to gain network connectivity and its associated MAP forwards its packets to the portal through the wireless multi-hop backhaul. The association between STAs and MAPs determines the logical network topology and has significant impact on network performances such as aggregate throughput and user fairness [3]. The default association metric adopted in the IEEE 802.11 standards is the received signal strength indicator, i.e. a STA associates with the MAP from which the received signal strength is the highest. This simple metric may result in congestion at the hotspot access points (APs) when the users are not uniformly located. Association control methods such as [4] have been proposed to balance the load among APs. However, in WMNs, it is not always good to do load balancing. Instead, it is preferred that more STAs associate with the MAPs that have larger backhaul capacity. The reason is that a backhaul packet transmission from the MAPs with good backhaul condition, which could be higher backhaul link rate or fewer number of hops away from the portal, requires less transmission time or fewer number of relays, and therefore consumes less network resource compared to transmitting the same packet from the poor-backhaul MAPs. On the other hand, if too many STAs associate with the good-backhaul MAPs, the access network of these MAPs could be over-congested. In this paper, we propose an optimization-based association control scheme as one component of the resource management framework, based on a realistic network model that considers both access and backhaul network transmission constraints.

The previous works on association control in WLANs [4, 5] or WMNs [3, 6–8], ignored the interference between neighboring AP or MAP cells by assuming perfect access network channel assignments. However, as more and more APs are deployed to support the fast growing Wi-Fi enabled mobile devices, the inter-cell interference becomes more and more inevitable. In the 2.4 GHz frequency band of the IEEE 802.11 standards, there are only 3 or 4 non-overlapping 20 MHz-wide channels and the number is 12 or 13 for the 5 GHz frequency band [9]. If considering new standard such like the 802.11ac that supports a channel bandwidth up to 160 MHz, the number of non-overlapping channels is even smaller. Many channel assignment schemes have been proposed for WLANs in the literature [9]. The objective of these schemes usually is to minimize the total interference experienced by either APs or STAs. However, in WMNs, as discussed above, it is preferred

that the good-backhaul MAPs carry more traffic. As a result, it makes sense to reduce the interference at these MAPs with priority so that they can accommodate more STAs. In this paper, we propose a channel assignment scheme as another component of the resource management framework, which iteratively improves the channel assignment using a metric of total-weighted-interference where the weight of a MAP is its traffic load.

We take the network in Fig. 1 as an example to illustrate the importance of joint association control and channel assignment. There are 3 MAPs, 4 STAs and 1 portal. The access link rates are labelled next to the links. STA S_1 , S_2 , and S_3 are associated with MAP M_1 , M_2 , and M_3 , respectively. STA S_4 needs to associate with either M_2 or M_3 . Suppose the backhaul links do not interfere with the access links and have enough bandwidth to support all the traffic of the STAs. Suppose there are a total of 2 non-overlapping channels $\{ch_1$ and $ch_2\}$ to be assigned to the MAPs, and at any time, only one of the MAPs assigned with the same channel can transmit or receive. Suppose the STAs on the same channel get equal transmission time. To see how association control makes a difference, we first consider a channel assignment in which M_1 , M_2 , and M_3 are assigned with ch_1 , ch_2 and ch_1 respectively. If S_4 is associated with M_2 , the aggregate throughput would be 57 Mbps ($54/2 + 36/2 + 18/2 + 6/2$). On the other hand, if S_4 is associated with M_3 , the aggregate throughput would be 62 Mbps ($54/3 + 36 + 18/3 + 6/3$). Next we illustrate how channel assignment makes a difference by changing the channel of M_3 from ch_1 to ch_2 . Now the aggregate throughput is further increased to 74 Mbps ($54 + 32/3 + 18/3 + 6/3$). Later we will see how our proposed channel assignment algorithm gets the second channel assignment from the first one.

The last essential component of our resource management framework is a utility-maximization-based STA bandwidth allocation algorithm. In the previous example, we only compare the aggregate throughput without considering user fairness. Network aggregate throughput and user fairness in bandwidth are usually two conflicting objectives in bandwidth allocation algorithms [10]. For example, in the example network in Fig. 1, we can get a maximum aggregate throughput of 90 Mbps by allowing only S_1 and S_2 to transmit and starving S_3 and S_4 , which is obviously extremely unfair to S_3 and S_4 . The IEEE 802.11 a/b/g MAC protocols implicitly enforce max-min throughput fairness among the users within one cell in the long term, i.e. each user gets equal transmission opportunity and achieves equal throughput [11]. Researchers have proposed other definitions of fairness such as the proportional fairness [12] and the time-based fairness [13] to make better use of the network resource. Instead of targeting at any single type of fairness as above, our bandwidth allocation algorithm achieves a utility-based fairness which is flexible in adjusting the trade-off between resource utilization efficiency and user fairness.

The rest of the paper is organized as follows. Section 2 introduces the related works and highlights the unique features of our proposed scheme. In Sect. 3, we present our network

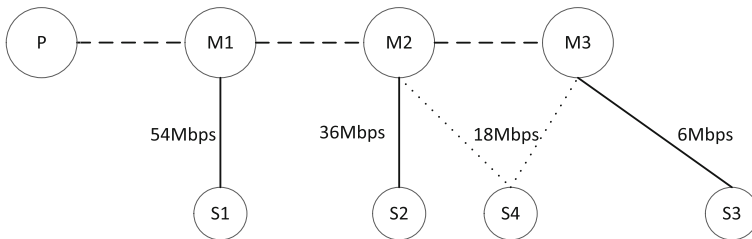


Fig. 1 An association control and channel assignment example

model of WMNs. In Sect. 4, we present algorithm details of our network resource management framework for WMNs. Performance evaluation via simulations is given in Sect. 5. Finally, Sect. 6 concludes the paper.

2 Related Works

The association control methods designed for WMNs in the literature are mostly heuristic-based, such as [3, 6], where each STA makes association decision based on the association metric of a MAP, which is a weighted sum of the estimated packet transmission time at the access network and the backhaul network of the MAP. There are very few optimization-based association control methods for WMNs. In [7], an optimal joint association, routing, and max-min bandwidth allocation problem is formulated, but the authors still adopt a heuristic association and routing approach to overcome the NP-hard problem, which makes the solution much less optimal. [8] has not only formulated the optimal association problems in WMNs but also proposed approximation algorithms together with theoretical analysis on the approximation ratio. Unlike in WMNs, there are more works on the optimal association control in WLANs. [4] and [5, 14] propose algorithms to find approximately optimal association achieving max-min fairness and proportional fairness, respectively. They first relax the integral association constraint and find an optimal fractional association where each STA is allowed to associate with multiple APs. Then the fractional association is rounded to an integral one through a bipartite-graph-based rounding algorithm. [15] formulates a joint association control, rate control, and contention resolution optimization problem and provides an approximation algorithm to solve the NP-hard problem. [16] further improves [15] by incorporating channel assignment into the problem formulation.

Most of the above mentioned association control methods, except [15, 16] for WLANs, assume carefully planned networks such that there is no interference between adjacent cells, which is rarely the case in reality. Our association control algorithm adopt a relaxation-and-rounding approach similar to that in [4, 5], but we take into consideration the WMN wireless multi-hop backhaul constraint as well as the access network interference between cells operating on the same channel. As far as we know, we are the first to investigate optimal association control in WMNs taking inter-MAP interference and access network channel assignment into consideration.

To the best of our knowledge, there are no previous works on the access network channel assignment problem in WMNs. The research works for WMNs have been focusing on the channel and radio assignment for the backhaul links [17]. On the other hand, the access network channel assignment problem has been well studied for WLANs [9]. Least Congested Channel Search (LCCS) [18] is a widely used channel selection method in the current WLANs, in which each AP autonomously searches every channel and switches to operate on the channel with the fewest number of STAs or the least amount of traffic. In [19], each AP locally measures the interference power experienced on every channel and switches to operate on a random channel according to a switching probability that is computed based on an annealed Gibbs sampler technique. Graph colouring is another classic approach for channel assignment. In [20], a heuristic vertex colouring algorithm using the “degree of saturation” is introduced, where a vertex with the largest number of differently coloured neighbours is chosen to be coloured in each iteration. [21] is a client-driven approach where the algorithm repeatedly assigns each AP a channel such that the

number of conflict-free clients is maximized until that number cannot be improved any more.

Instead of using the objectives such as the interference at the APs, the number of coloured APs, or the number of conflicting clients as above that are implicitly related to clients' throughput, our channel assignment algorithm repeatedly invokes an optimal bandwidth allocation procedure to iteratively improve the utility objective function value which explicitly counts the bandwidth of each client. In addition, in the process of finding the best channel for each MAP, we make use of a metric named "total-weighted-interference" that takes into consideration the load carried by the MAP; through that, the interference at the good-backhaul MAPs can be reduced and the network capacity of the entire WMN can be improved.

3 Network Model

We consider an infrastructure WMN that consists of STAs, MAPs, and a portal. One MAP has two interfaces: one is the access interface that communicates with the STAs and performs the AP functionality; the other is the backhaul interface that participates in the formation of the wireless multi-hop backhaul. The portal connects the WMN with other networks. We consider Internet traffic only, i.e. the STAs send (receive) packets to (from) the Internet through the portal, as low-cost Internet access is the most common usage of a WMN. All backhaul links operate on the same channel to maintain connectivity, while the MAP access interfaces operate on different channels to reduce the inter-cell interference. We assume that the backhaul links do not interfere with the access links, which can be realized by letting them adopt different 802.11 standards or operate on different channels.

We assume that the wireless multi-hop backhaul routing graph is a tree topology with the root at the portal. Such routing can be done by a proactive routing protocol such as Hybrid Wireless Mesh Protocol (HWMP) working in the proactive mode [22], in which each MAP finds its shortest path towards the portal. We assume that we know the transmission rate of the MAP–MAP backhaul links and the MAP–STA access links.

We adopt a protocol-based transmission model where each node has a fixed transmission range and a fixed interference range. To take account of both upstream and downstream traffic, we consider two links conflicting with each other if either end of one link is in the interference range of either end of the other link. Then we can construct a conflict graph for the backhaul links and conflict graphs for the access links on different channels.

We make use of the concept of "clique of links" to model the concurrent transmission constraint of links. A clique is a set of links that are in mutual conflict with each other, i.e. at any time, only a single link within a clique is allowed to transmit. With the constructed conflict graphs, we can find all the maximal cliques in the network using algorithms such as the Bron–Kerbosch algorithm [23]. However, finding all the maximal cliques over the entire network is NP-hard and exponential-time algorithms are not scalable for large networks with thousands of links. Therefore we propose an efficient and effective heuristic clique modeling method that is able to find a set of cliques that are very close to the true set of maximal cliques. Our method looks for cliques locally in the subsets of links that are carefully selected. For each subset of links, we apply the Bron–Kerbosch algorithm to find local-maximal cliques. We will see in the simulation results in Sect. 5 that the performance of the proposed method is almost identical to that of the exponential-time algorithm.

Our method is efficient because the size of any subset of links in our model is significantly smaller than the total number of links in the network, which is true under our assumption that the number of links that conflict with any given link is smaller than the total number of links in the network. The assumption is reasonable because our method targets at WMNs that are large enough to accommodate a multi-hop backhaul and in such network it is usually possible to have concurrent transmission and not all the backhaul links conflicting with any given link. When a WMN is small and the total number of links is not very large, applying exponential-time algorithms over the entire network will not consume extremely long time and heuristic methods like ours is not necessary.

We take the network in Fig. 2 as an example to illustrate how our clique modeling method works. The set of backhaul links is $\{l_1, l_2, l_3, l_4\}$. The backhaul links have equal length. The interference range of the MAPs, in the backhaul, is longer than the one-hop distance and shorter than the two-hop distance. The set of all the maximal backhaul-link cliques in the network is $MC_B = \{\{l_1, l_2, l_3\}, \{l_2, l_3, l_4\}\}$. In our modeling framework, taking link l_2 for example, we first find the set of links that are in conflict with link l_2 : $L_{cfl}(l_2) = \{l_1, l_3, l_4\}$. Then we find cliques locally among the links in $L_{cfl}(l_2)$: $K_{cfl}(l_2) = \{\{l_1, l_3\}, \{l_3, l_4\}\}$. Including l_2 itself, we get a set of local cliques at l_2 : $k(l_2) = \{\{l_1, l_2, l_3\}, \{l_2, l_3, l_4\}\}$. Combining $k(l_2)$ with the local cliques at l_1, l_3 , and l_4 , we get a set of backhaul-link cliques K_B that is identical with MC_B .

The set of access links in Fig. 2 is $\{l_{11}, l_{15}, l_{22}, l_{33}, l_{44}\}$. MAP M_1, M_2 , and M_3 operate on ch_1 , while M_4 operates on ch_2 . STA S_5 is in the interference range of M_3 , while S_1 is not. The set of all the maximal access-link cliques in the network is $MC_A = \{\{l_{11}, l_{15}, l_{22}\}, \{l_{15}, l_{22}, l_{33}\}, \{l_{44}\}\}$. In our modeling framework, taking MAP M_3 for example, we first find a set of interfering MAPs of M_3 : $M_{if}(M_3) = \{M_1, M_2\}$. Then we find a set of MAP-cliques of M_3 among the MAPs in $M_{if}(M_3)$: $Q_{if}(M_3) = \{\{M_1, M_2\}\}$. For the MAP-clique $q = \{M_1, M_2\}, q \in Q_{if}(M_3)$, we generate an access-link clique $k(M_3, q) = \{l_{15}, l_{22}, l_{33}\}$. Combining the access-link cliques generated from the MAP-cliques in $Q_{if}(M_3)$, we get a set of access-link cliques at M_3 : $k(M_3) = \{\{l_{15}, l_{22}, l_{33}\}\}$. Combining $k(M_3)$ with the cliques at M_1, M_2 , and M_4 , we get a set of access-link cliques K_A that is identical with MC_A . Next we give the details of our clique modeling method.

Backhaul-link Cliques Given the set of the backhaul links L_B and the backhaul conflict graph (BCG), we can find a set of backhaul-link cliques by finding a set of cliques locally for each link $l \in L_B$. We first find the set of links that are in conflict with link l :

$$L_{cfl}(l) = \{l' : l' \in L_B \wedge BCG(l, l') = 1\},$$

where $BCG(l, l') = 1$ means link l and l' are in conflict with each other. Then we can find cliques among the links in $L_{cfl}(l)$ using the Bron–Kerbosch algorithm:

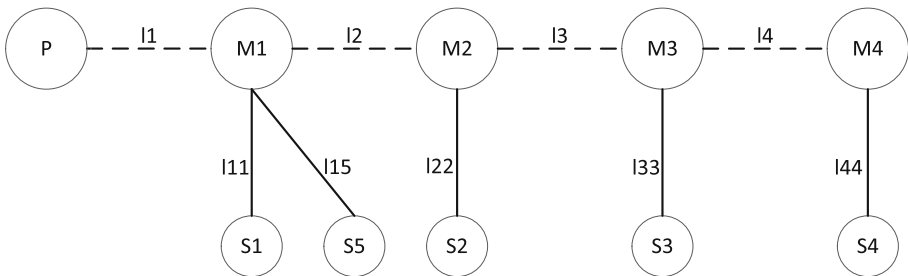


Fig. 2 A clique modeling example

$$K_{cfl}(l) = \{k = \{l'\} : \forall l'_1, l'_2 \in k : l'_1, l'_2 \in L_{cfl}(l) \wedge l'_1 \neq l'_2 \wedge BCG(l'_1, l'_2) = 1\}.$$

Including link l itself, we get a set of local cliques at l :

$$k(l) = \{\{l \cup k\} : k \in K_{cfl}(l)\}.$$

After combining and simplifying the local cliques of all the backhaul links, we finally get a set of backhaul-link cliques

$$K_B = \{k(l) : l \in L_B\}.$$

Access-Link Cliques Given the set of MAPs M , the set of STAs S , the transmission range $TransR$, the interference range $IntR$, and the locations of the MAPs and STAs, we can find a set of access-link cliques by finding a set of MAP-cliques among the interfering MAPs of each MAP $i \in M$. We first find the set of MAPs that are within the interference range of i :

$$M_{IR}(i) = \{i' : i' \in M \wedge dis(i', i) < IntR\},$$

the set of STAs that are within the interference range of i :

$$S_{IR}(i) = \{j : j \in S \wedge dis(j, i) < IntR\},$$

and the set of STAs that are within the transmission range of i :

$$S_{TR}(i) = \{j : j \in S \wedge dis(j, i) < TransR\},$$

where $dis(\cdot)$ is the distance between the two nodes. Given a channel assignment $C = \{c_i\}$, where c_i is the channel assigned to MAP $i \in M$, we can find two sets of interfering MAPs of i . One is the set of MAPs that are within the interference range of i :

$$M_{if,1}(i) = \{i' : i' \in M \wedge c_{i'} = c_i \wedge i' \in M_{IR}(i)\};$$

the other is the set of MAPs that are outside the interference range of i , but have access links interfering with i :

$$M_{if,2}(i) = \{i' : i' \in M \setminus M_{IR}(i) \wedge c_{i'} = c_i \wedge (\exists j : j \in S_{TR}(i') \wedge j \in S_{IR}(i))\}.$$

Combining the two, we get

$$M_{if}(i) = M_{if,1}(i) \cup M_{if,2}(i).$$

Applying the Bron–Kerbosch algorithm among the MAPs in $M_{if}(i)$, we can find a set of MAP-cliques of i :

$$Q_{if}(i) = \{q = \{i'\} : \forall i'_1, i'_2 \in q : i'_1, i'_2 \in M_{if}(i) \wedge i'_1 \neq i'_2 \wedge i'_1 \in M_{IR}(i'_2) \wedge i'_2 \in M_{IR}(i'_1)\}.$$

For each MAP-clique $q \in Q_{if}(i)$, we generate an access-link clique:

$$k(i, q) = \{l_{ij} : j \in S\} \cup \{l_{ij} : i' \in q \cap M_{if,1}(i), j \in S\} \cup \{l_{ij} : i' \in q \cap M_{if,2}(i), j \in S_{TR}(i)\},$$

where l_{ij} is the access link between MAP i and STA j . Then we can get a set of local access-link cliques at i :

$$k(i) = \{k(i, q) : q \in Q_{\text{if}}(i)\}.$$

After combining and simplifying the local access-link cliques of all the MAPs, we finally get a set of access-link cliques

$$K_A = \{k(i) : i \in M\}.$$

Table 1 summarizes some of the notations used in the paper. We use y_{ki} to indicate whether MAP i 's backhaul path passes through the backhaul clique k , i.e. $y_{ki} = 1$ if there exists a link $l \in k$ such that $l \in \text{path}(i)$, where $\text{path}(i)$ is the set of links that are on the routing path between i and the portal. If $y_{ki} = 1$, we use r_{ki} to denote the effective backhaul link rate for i in k . r_{ki} is defined in (1), which represents the time consumed for transmitting one bit of i 's traffic in k .

$$\frac{1}{r_{ki}} = \sum_{l:l \in k \wedge l \in \text{path}(i)} \frac{1}{r_l} \tag{1}$$

We use x_{ij} to indicate the association between MAP i and STA j . $x_{ij} \in \{0, 1\}$ for an integral association where each STA is allowed to associate with only one MAP. $x_{ij}=1$ if j is associated with i ; otherwise 0. $x_{ij} \in [0, 1]$ for a fractional association where each STA is allowed to fractionally associate with multiple MAPs. In both cases, we have $\forall j \in S : \sum_{i \in M} x_{ij} = 1$. We use b_j to denote the bandwidth allocated to j , and b_{ij} to denote the bandwidth allocated to j communicating with i in a fractional association. So we have $b_j = \sum_{i \in M} b_{ij}$ and $x_{ij} = b_{ij}/b_j$. The outcome of our resource management framework is a channel assignment vector $\{c_i\}$, an association matrix $\{x_{ij}\}$, and a bandwidth allocation vector $\{b_j\}$, which is denoted as (C, X, B) .

Table 1 Notations

Symbols	Semantics
M	The set of all MAPs $\{i\}$
S	The set of all STAs $\{j\}$
CH	The set of all non-overlapping channels $\{ch_1, ch_2, \dots, ch_{N-CH}\}$
X	A STA-MAP association matrix $\{x_{ij}\}$
B	A STA bandwidth allocation vector $\{b_j\}$
C	A channel assignment vector $\{c_i\}$
K_B	The set of all backhaul-link cliques
y_{ki}	Indicating whether clique k is on MAP i 's backhaul path
r_{ki}	The effective backhaul link rate of MAP i in clique k
r_{ij}	The access link rate between MAP i and STA j
b_{ij}	The bandwidth allocated to STA j to communicate with MAP i
$M_{\text{if}}(i)$	The set of MAPs that have access links interfering with MAP i on the same channel
$Q_{\text{if}}(i)$	The set of maximal MAP-cliques among the MAPs in $M_{\text{if}}(i)$

4 A Network Resource Management Framework for WMNs

Our resource management framework consists of three components: bandwidth allocation, channel assignment, and association control. First, we formulate the optimal utility fair bandwidth allocation problems. Then we introduce a joint channel assignment and bandwidth allocation algorithm. Finally, we present an optimization-based association control scheme as well as the complete resource management framework.

4.1 Utility-Based Bandwidth Allocation

The objective of our bandwidth allocation algorithm is to maximize the sum of the utility of user bandwidth. The utility function we use is given in (2) and it was proposed in [24], where b_j is the bandwidth of STA j , and α is the parameter controlling the trade-off between resource efficiency and user fairness. When $\alpha = 0$, the objective is to maximize the network throughput, with no consideration in user fairness. As α increases, the aggregate throughput decreases and the bandwidth allocation becomes fairer and fairer. When $\alpha = 1$, the objective is proportional fairness. When α approaches infinity, absolute fairness dominates and max-min fairness is the objective.

$$U_\alpha(b_j) = \begin{cases} \log b_j & \text{if } \alpha = 1 \\ (1 - \alpha)^{-1} b_j^{1-\alpha} & \text{if } \alpha \neq 1 \end{cases} \tag{2}$$

Our bandwidth allocation algorithm is named Utility-based Bandwidth allocation (UBa) and elaborated in Fig. 3. Given the network topology and a channel assignment C , we can construct the set of backhaul-link cliques, K_B , and the set of local maximal cliques of the interfering MAPs, $Q_{if}(i)$, for each MAP i . If we are given an integral association matrix X , we formulate an optimization problem named Integral Problem (IntP) as in (3), where each STA is allowed to associate with one MAP only. By solving IntP, we get an integral bandwidth allocation vector $\{b_j\}$. On the other hand, if the integral association is unknown, we formulate another optimization problem named Fractional Problem (FracP) as in (4), where each STA is allowed to fractionally associate with multiple MAPs. By solving FracP, we get a fractional bandwidth allocation matrix $\{b_{ij}\}$.

$$\text{IntP :} \quad \text{Max} \quad \sum_{j \in S} U_\alpha(b_j) \tag{3a}$$

$$\text{s.t.} \quad \forall j \in S : \sum_{i \in M} x_{ij} = 1 \tag{3b}$$

$$\forall j \in S : \sum_{i \in M} x_{ij} \frac{b_j}{r_{ij}} \leq 1 \tag{3c}$$

$$\forall k \in K_B : \sum_{i \in M} \frac{y_{ki}}{r_{ki}} \sum_{j \in S} x_{ij} b_j \leq 1 \tag{3d}$$

Algorithm UBa

Given: the network topology, the backhaul routing tree, the backhaul link set L_B
 Input: (C, X)

1. Construct a backhaul link conflict graph BCG
2. Find $L_{cfl}(l)$ and $Q_{cfl}(l)$ for each link $l \in L_B$
3. Find $K_B, \{y_{ki}\}, \{r_{ki}\}$
4. **for** each MAP $i \in M$ **do**
 - (a) Find $M_{IR}(i), S_{IR}(i), S_{TR}(i)$
 - (b) Find $M_{itf,1}(i), M_{itf,2}(i),$ and $M_{itf}(i)$
 - (c) Construct a conflict graph for MAPs in $M_{itf}(i)$
 - (d) Find $Q_{itf}(i)$**end for**
5. **if** $X = 0$ **then**

Formulate and solve the problem FracP
return the fractional bandwidth allocation solution $\{b_{ij}\}$

else

Formulate and solve the problem IntP
return the integral bandwidth allocation solution $\{b_j\}$

end if

Fig. 3 Algorithm UBa

$$\forall q \in Q_{itf}(i), i \in M : \sum_{j \in S} x_{ij} \frac{b_j}{r_{ij}} + \sum_{i': i' \in q \wedge i' \in M_{IR}(i)} \sum_{j \in S} x_{i'j} \frac{b_j}{r_{i'j}} + \sum_{i': i' \in q \wedge i' \notin M_{IR}(i)} \sum_{j \in S_{IR}(i)} x_{i'j} \frac{b_j}{r_{i'j}} \leq 1 \tag{3e}$$

$$\forall i \in M, j \in S : x_{ij} \in \{0, 1\}, b_j \geq 0 \tag{3f}$$

$$\text{FracP : Max } \sum_{j \in S} U_z(b_j) \tag{4a}$$

$$\text{s.t. } \forall j \in S : b_j = \sum_{i \in M} b_{ij} \tag{4b}$$

$$\forall j \in S : \sum_{i \in M} \frac{b_{ij}}{r_{ij}} \leq 1 \tag{4c}$$

$$\forall k \in K_B : \sum_{i \in M} \frac{y_{ki}}{r_{ki}} \sum_{j \in S} b_{ij} \leq 1 \tag{4d}$$

$$\forall q \in Q_{itf}(i), i \in M : \sum_{j \in S} \frac{b_{ij}}{r_{ij}} + \sum_{i': i' \in q \wedge i' \in M_{IR}(i)} \sum_{j \in S} \frac{b_{i'j}}{r_{i'j}} + \sum_{i': i' \in q \wedge i' \notin M_{IR}(i)} \sum_{j \in S_{IR}(i)} \frac{b_{i'j}}{r_{i'j}} \leq 1 \tag{4e}$$

$$\forall i \in M, j \in S : b_{ij} \geq 0 \tag{4f}$$

Constraint (3b) states that each STA is associated with one MAP only. (3c) states that the total transmission time of one STA is less than the unit time. (3d) states that the total transmission time of the backhaul links in one backhaul clique is less than 1, where the

traffic load carried by the clique originates from all the STAs whose associated MAP backhaul paths towards the portal pass through the clique. (3e) states that the total transmission time of the access links belonging to one access clique is less than 1. By introducing the fractional bandwidth allocation matrix $\{b_{ij}\}$, FracP is derived from IntP by replacing b_j with $\sum_{i \in M} b_{ij}$ and replacing $x_{ij} \cdot b_j$ with b_{ij} .

4.2 Joint Channel Assignment and Bandwidth Allocation

We propose a channel assignment scheme, named Joint Channel assignment and Bandwidth allocation (JCaBa), which iteratively improves channel assignment and network performance by invoking the bandwidth allocation algorithm UBa and making use of an interference metric that measures the interference experienced by a MAP as well as the interference caused by the MAP to the others. Algorithm JCaBa is given in Fig. 4.

Given a channel assignment C , we can get an optimal fractional bandwidth allocation matrix $B = \{b_{ij}\}$ by applying algorithm UBa with input $(C, 0)$. Denote the utility objective function value of B as $f_{utility}(B)$. For each MAP $i \in M$, denote the total traffic to be carried by i for its associated STAs as $load_{self}(i)$, i.e. self-load of i :

Algorithm JCaBa

```

 $C^* \leftarrow$  random channel assignment;  $B^* \leftarrow$  UBa( $C^*, 0$ );  $M_{allocated} \leftarrow \emptyset$ ;  $change \leftarrow 0$ 
 $Load^*_{self} \leftarrow$  CalculateLs( $B^*$ ) {sort  $Load^*_{self}$  in non-increasing order}
 $n \leftarrow 1$  {let  $i(n)$  represent the  $n$ th MAP in the sorted vector  $Load^*_{self}$ }
while  $n \leq |M|$  do
  if  $i(n) \notin M_{allocated}$  then
     $c_0$  is the current channel of  $i(n)$  in  $C^*$ 
    for  $c \in CH \setminus c_0$  do
      change the channel of  $i(n)$  to  $c$ , denote the new channel assignment as  $C_c$ 
       $load_c \leftarrow$  CalculateWI( $C_c, B^*, i(n)$ )
    end for
     $load_{min} \leftarrow \min\{load_c : c \in CH \setminus c_0\}$ , denote the corresponding channel assignment as  $C'$ 
     $B' \leftarrow$  UBa( $C', 0$ )
     $Load'_{self} \leftarrow$  CalculateLs( $B'$ )
    if  $f_{utility}(B') > f_{utility}(B^*)$  then
      {assign  $i(n)$  with the new channel}
       $C^* \leftarrow C'$ ;  $B^* \leftarrow B'$ ;  $Load^*_{self} \leftarrow Load'_{self}$ ;  $n \leftarrow 1$ ;  $change \leftarrow 1$ 
    else
       $n \leftarrow n + 1$  {keep the channel of  $i(n)$  unchanged}
    end if
    Add  $i(n)$  to  $M_{allocated}$ 
  else
     $n \leftarrow n + 1$  {the channel of  $i(n)$  has been assigned already}
  end if
if  $n = |M| + 1$  and  $change = 1$  then
   $n \leftarrow 1$ ;  $change \leftarrow 0$ ;  $M_{allocated} \leftarrow \emptyset$ 
end if
end while
return  $C^*$  and  $f_{utility}(B^*)$ 

```

Fig. 4 Algorithm JCaBa

$$load_{self}(i) = \sum_{j \in S} b_{ij} \tag{5}$$

For each MAP $i' \in M_{if}(i)$, i.e. i' is one of the interfering MAPs of i , denote the traffic that is carried by i' and interferes with i as $load_{if}(i, i')$, i.e. the interference-load to i from i' :

$$load_{if}(i, i') = \begin{cases} \sum_{j \in S} b_{ij} & \text{if } i' \in M_{if,1}(i) \\ \sum_{j \in S_{IR}(i)} b_{ij} & \text{if } i' \in M_{if,2}(i) \end{cases} \tag{6}$$

Denote the total self-load of i and its interfering MAPs in $M_{if}(i)$ as $load_{t-self}(i)$:

$$load_{t-self}(i) = load_{self}(i) + \sum_{i' \in M_{if}(i)} load_{self}(i') \tag{7}$$

We define a metric of total-weighted-interference for i , denoted as $load_{t-w-if}(i)$. The metric is formulated in (8) and consists of two parts. The first part is the total interference experienced by i , weighted by $load_{self}(i)/load_{t-self}(i)$. The second part is the total interference to MAPs in $M_{if}(i)$ caused by i , weighted by $load_{self}(i')/load_{t-self}(i)$ for each $i' \in M_{if}(i)$. By weighting a MAP with its self-load divided by the total load, the interference experienced by the heavy-self-load MAPs contributes more to the total interference. As introduced in Sect. 1, in WMNs, it is preferred that the good-backhaul MAPs carries more traffic load. Therefore, by reducing the total-weighted-interference metric defined in (8), we reduce the interference experienced by the good-backhaul MAPs and increase the network capacity.

$$load_{t-w-if}(i) = \frac{load_{self}(i)}{load_{t-self}(i)} \cdot \sum_{i' \in M_{if}(i)} load_{if}(i, i') + \sum_{i' \in M_{if}(i)} \frac{load_{self}(i')}{load_{t-self}(i)} \cdot load_{if}(i', i) \tag{8}$$

In Fig. 4, the function CalculateLs(B) calculates the self-load vector $Load_{self} = \{load_{self}(i)\}$ using (5). The function CalculateWI(C, B, i) calculates the total-weighted-interference $load_{t-w-if}(i)$ for MAP i using (8), when the channel assignment is C and the bandwidth allocation is B . We sort the MAPs in decreasing order of their self-load so that the MAPs carrying heavier load are assigned channels first. In one round of the while-loop, the sorted MAPs, one by one from the beginning, decide to stay in the current channel or switch to a new channel. In order to make that decision, a MAP first searches for the channel with the least total-weighted-interference; denote the corresponding channel assignment and bandwidth allocation as C' and B' , respectively, while the current channel assignment and bandwidth allocation are C^* and B^* . If the objective function value of the new bandwidth allocation, $f_{utility}(B')$, is larger than that of B^* , the MAP switches to the new channel; otherwise, it stays with the current channel. The algorithm terminates if no MAP switches channel in the last round of the while-loop.

As a channel change is allowed only if that change can improve the global objective function value, the objective function value monotonically increases while the algorithm is executed. Because the objective function value is upper bounded by the optimal solution, the JCaBa algorithm will converge eventually. At the time of convergence, no MAP can find a new channel assignment that can improve the network performance globally. The convergence speed depends on the number of MAPs affected by one channel change. If the MAP density is low or the number of orthogonal channels in the network is large, a small number of channel changes will occur in each while-loop and the algorithm will converge

fast. In our simulations, the algorithm always converges within few rounds of the while-loop.

Let us revisit the network in Fig. 1. Suppose we are given a channel assignment $C_0 = \{ch_1, ch_2, ch_1\}$ that is produced by a vertex colouring algorithm. We demonstrate how the JCaBa algorithm improves C_0 and finds a better channel assignment. With C_0 , the best aggregate throughput is 62 Mbps that is obtained when S_4 associates with M_3 , and the corresponding self-load vector is $\{18, 36, 8\}$. In the first round of the while-loop: M_2 examines $\{ch_1, ch_1, ch_1\}$, for which the aggregate throughput is 28.5 Mbps, and decides to stay with ch_2 ; then M_1 examines $\{ch_2, ch_2, ch_1\}$, for which the best aggregate throughput is 57 Mbps, and decides to stay with ch_1 ; finally, M_3 examines $\{ch_1, ch_2, ch_2\}$, for which the aggregate throughput is 74 Mbps, and decides to switch to ch_2 . In the second round of the while-loop, as no MAP can find a better channel assignment, the algorithm terminates and returns the channel assignment $\{ch_1, ch_2, ch_2\}$.

4.3 The Resource Management Framework for WMNs

We name our resource management framework Joint Channel assignment, Bandwidth allocation and Association control algorithm (JCBA), which is given in Fig. 5. It has been shown in [25] that channel assignment should be conducted prior to association control for better network performance. It makes sense as channel assignment determines the interference between cells at a large scale and should be performed less frequently compared to association control. Therefore, the first step of JCBA is to determine a proper channel assignment by applying the JCaBa algorithm a few times with different initial random channel assignments. The reason to do that is JCaBa locally searches for better channels and may not generate a global optimal channel assignment in a single run. The channel assignment that achieves the largest objective function value is selected and denoted as C' .

The second step of JCBA is to generate an integral association, denoted as \hat{X}' , by applying an association control algorithm, named optimization-based Association Control (oAC). The first step of oAC is to find a fractional bandwidth allocation, B' , by applying the UBa algorithm with $(C', 0)$ as input. Then we convert B' to a fractional association matrix X' according to the equation

$$x'_{ij} = b'_{ij} / \sum_{i \in M} b'_{ij}.$$

Algorithm JCBA

1. Channel Assignment: Apply the JCaBa algorithm a few times and select the best channel assignment C' .
2. Association Control: Generate an integral association \hat{X}' by applying the oAC algorithm:
 - (a) Get a fractional bandwidth allocation B' using the algorithm $UBa(C', 0)$.
 - (b) Generate a fractional association X' from B' .
 - (c) Round X' to \hat{X}' by the randomization rounding.
3. Bandwidth Allocation: Get an integral bandwidth allocation \hat{B}' using the algorithm $UBa(C', \hat{X}')$.

(C', \hat{X}', \hat{B}') is the output

Fig. 5 Algorithm JCBA

In the third step of oAC, X' is rounded to the integral solution \hat{X}' , via randomization rounding. Denote the set of MAPs that have fractional association with STA j in X' as M_j , i.e. $M_j = \{i : b'_{ij} > 0\}$. By the randomization rounding, j randomly selects one of the MAPs in M_j to associate with.

In the final step of JCBA, we get an integral bandwidth allocation, denoted as \hat{B}' , by applying the UBa algorithm with (C', \hat{X}') as input. Finally, (C', \hat{X}', \hat{B}') is the output of the JCBA algorithm.

JCBA is a centralized optimization-based resource management framework. Centralized algorithms generally outperform distributed algorithms as the central controller is aware of the whole network condition and optimization algorithms can be implemented. In the centralized algorithms, it is assumed that the central controller is aware of the entire network topology as well as the achievable link rates. It is also assumed that these information can be shared with the central controller by the MAPs and STAs efficiently and reliably.

The MAPs in WMNs usually have minimal mobility and form a relatively stable multi-hop wireless backhaul. Therefore the location information of the deployed MAPs can be assumed to be available through site maps or site survey. When a STA joins the network, it will first scan each channel and receive beacon messages from the MAPs in the vicinity. If the STA reports the received signal strength of the beacons of the nearby MAPs to the central controller, the central controller may infer the location of the STA through any localization techniques. With the location information of the MAPs and STAs, the central controller can construct a protocol-based network model and an interference map to model the concurrent transmission constraints in the network.

For centralized algorithms, we would expect more overhead than in the distributed algorithms as the MAPs and STAs need to report their link rates, received signal strength in the vicinity, and association status to the central controller and the central controller needs to distribute its control messages to the MAPs. There will be overhead due to control message exchange. Depending on how fast the network condition changes, the amount of the overhead would be different. If the network is stable, the amount of control messages would be small.

Besides overhead, another concern with centralized algorithms is computational complexity. In the JCBA algorithm, the most time consuming step is executing the channel assignment algorithm, JCaBa. The good news is that instead of frequently conducting JCaBa, most of the time we only need to conduct the association control algorithm (oAC) for three reasons. Firstly, a channel switch at a MAP incurs channel switching at all of its associated STAs, while an association change requires action at one STA only, i.e. invoking JCaBa interrupts more users. Secondly, oAC is much more effective in improving the network performance than JCaBa, which will be demonstrated by our simulation results. Thirdly, oAC is more time efficient as only one convex problem needs to be solved.

Though the oAC algorithm is time efficient, it should be triggered only when it is necessary to do so. Excessive triggering would cause unnecessary interruption to normal communication and should be avoided. On the other hand, inadequate triggering may miss network dynamics and result in inferior performance. oAC should be triggered when the network condition has significantly changed, e.g. mass joining/leaving of MAPs/STAs, or blocked/broken backhaul paths. oAC should not be triggered by minor network changes such as a single link rate change. The triggering mechanism could be periodic-time-based or based on real time network condition measurement.

5 Performance Evaluation

5.1 Simulation Setting

We present simulation results for a WMN that consists of 20 MAPs, 100 STAs, and 1 portal. The MAPs are randomly placed in a square field of size 250 m × 250 m. The portal is located at the centre of the lower-left quadrant. We simulate two user topologies: uniform topology where the STAs are randomly placed in the field; hotspot topology where the STAs are distributed in two randomly located hotspots of radius 50 m each. We provide network topology examples for the two user topologies in Fig. 6, where red diamonds, big green circles, and small yellow circles represent the portal, the MAPs, and the STAs, respectively; the backhaul links are displayed in red lines.

Assuming a transmitter power of 17dBm, a receiver noise power of -80dBm, a Clear Channel Access (CCA) threshold of -76dBm, and adopting a log-distance path loss model, we simulate a protocol-based network model with a transmission range of 100 m and an interference range of 150 m using Matlab. An access link rate model is constructed and given in Table 2, where the required minimum Signal-to-Noise Ratio (SNR) is taken from [26]. We simulate a WMN such that the access networks operate on 4 non-overlapping 20 MHz channels in the 2.4 GHz frequency band, while the backhaul network operates on a single channel in the 5 GHz frequency band. As the wireless backhaul carries the aggregate traffic of the entire network and the MAPs are more powerful than the STAs, the backhaul link rates are set to be 16 times of the access link rates, which can be achieved by applying more spatial streams and adopting a wider-bandwidth backhaul channel.

The backhaul routing tree rooted at the portal is constructed using the 802.11s HWMP routing protocol. We model the transmission constraints at the backhaul and the access networks using the local-clique-based modeling methods introduced in Sect. 3. We have done simulation with other configurations, such as different MAP/portal topologies,

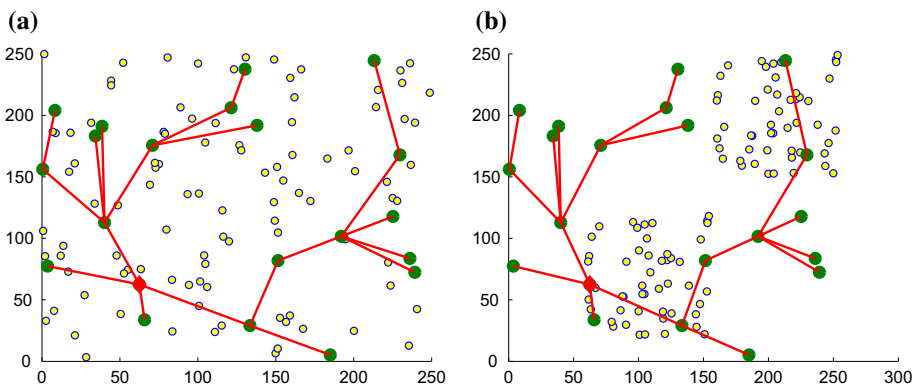


Fig. 6 Network topology examples. **a** Uniform topology, **b** Hotspot topology

Table 2 Link rate model for access links

Rate (Mbps)	6	12	18	24	36	48	54	60
Min. SNR (dB)	5	7	9	13	17	20	22	23
Max. path length (m)	100	89	79	63	50	42	38	35

different number of STAs, and different backhaul/access link rate ratios; their results are qualitatively similar to those we are presenting.

We measure the performance of different algorithms in terms of aggregate throughput, per-user bandwidth, and Jain’s fairness index [27]

$$F = \left(\sum_{j \in S} b_j \right)^2 / \left(|S| \cdot \sum_{j \in S} b_j^2 \right).$$

The index measures the fairness of a bandwidth allocation $\{b_j\}$ and ranges from 0 to 1. F equals 1 when all STAs have equal bandwidth and decreases as the bandwidth vector deviates from the ideal equal-bandwidth vector.

5.2 Performance of the Local-Clique-Based Modeling Method

A backhaul clique modeling method finds the set of backhaul-link cliques, K_B , which is required for formulating the backhaul transmission constraint (3d) and (4d) in the bandwidth allocation algorithm UBa. We compare the performance of the following backhaul clique modeling methods:

- *local-BC* our heuristic backhaul-link clique modeling method that constructs K_B by finding a set of backhaul-link cliques locally at each link $l \in L_B$.
- *o-BMC* optimal clique modeling method that finds all the backhaul maximal cliques in the network in exponential time.
- *a-BC* a heuristic clique modeling method used in [7] that approximates a backhaul maximal clique by the set of conflicting links of a backhaul link, i.e.

$$K_B = \{ \{L_{cl}(l) \cup l\} : l \in L_B \}.$$

An access clique modeling method finds a set of access-link cliques for each channel, which is required for formulating the access network transmission constraints in UBa. We compare the performance of the following access clique modeling methods:

- *local-AC* our heuristic access-link clique modeling method that constructs the access transmission constraints (3e) and (4e) by finding a set of MAP-cliques locally at each MAP $i \in M$.
- *o-AMC* optimal clique modeling method that finds all the access maximal cliques in the network in exponential time. Accordingly, constraints (3e) and (4e) are replaced by (9a) and (9b), respectively, where $K_A(c)$ is the set of all access cliques on channel c .
- *a-AC* a heuristic access clique approximation method that approximates an access clique by the links of the MAPs interfering with a MAP, i.e.

$$K_A = \{ \{ \{I_{ij}\} \cup \{I_{i'j} : i' \in M_{IR}(i) \wedge c_{i'} = c_i\} \} : i \in M \}.$$

Accordingly, constraints (3e) and (4e) are replaced by (10a) and (10b), respectively.

$$\forall k \in K_A(c), c \in CH : \sum_{l_{ij} \in k} x_{ij} \frac{b_j}{r_{ij}} \leq 1 \tag{9a}$$

$$\forall k \in K_A(c), c \in CH : \sum_{l_{ij} \in k} \frac{b_{ij}}{r_{ij}} \leq 1 \tag{9b}$$

$$\forall i \in M : \sum_{j \in S} x_{ij} \frac{b_j}{r_{ij}} + \sum_{i': i' \in M_{IR}(i) \wedge c_{i'} = c_i} \sum_{j \in S} x_{i'j} \frac{b_j}{r_{i'j}} \leq 1 \tag{10a}$$

$$\forall i \in M : \sum_{j \in S} \frac{b_{ij}}{r_{ij}} + \sum_{i': i' \in M_{IR}(i) \wedge c_{i'} = c_i} \sum_{j \in S} \frac{b_{i'j}}{r_{i'j}} \leq 1 \tag{10b}$$

Figure 7 depicts the bandwidth allocation results of the UBa algorithm with different transmission constraints that are obtained from the clique modeling methods. The results presented are averaged over 50 runs. In each run, the STA location is different and we sort the STAs in non-decreasing order of their allocated bandwidth. So the bandwidth of a STA indexed x in the figure indicates the average bandwidth of the x -th lowest bandwidth in each run. We compare the backhaul (access) clique modeling methods, while using o-AMC (o-BMC) as the access (backhaul) clique modeling method. The α parameter in the UBa algorithm is set to 5. As o-BMC and o-AMC are able to find all the maximal cliques in the network, their performance results are the benchmark for the other two heuristic methods. The closer a modeling method’s performance is to the benchmark, the better the method is.

In Fig. 7a, local-BC and o-BMC have identical results, indicating that local-BC is able to find all the backhaul maximal cliques in the simulated networks. In contrast, the results of a-BC clearly deviate from the benchmark. This is because in a-BC, links that are in conflict with the same link but do not interfere with each other are prohibited from concurrent transmission, while they are able to do so actually.

In Fig. 7b, the curve of local-AC almost coincides with that of o-AMC. The reason for the small difference is that local-AC is based on MAP-cliques and may miss few links that should have been included in the access-link cliques, resulting in a slightly loosened access network transmission constraint. The results of a-AC seriously deviate from the benchmark. a-AC prohibits MAPs that interfere with one common MAP but do not interfere with each other from concurrent transmission, even though that would not cause any collision.

o-BMC and o-AMC search for maximal cliques over the entire network using exponential time algorithms and can be very time consuming for large networks. On the other hand, our local-clique-based methods, local-BC and local-AC, locally search for cliques where the number of variables is limited and much smaller. In addition, as seen in Fig. 7, our methods achieve performance that is very close to the benchmark. Therefore, we can say that the local-clique-based modeling methods are efficient and effective.

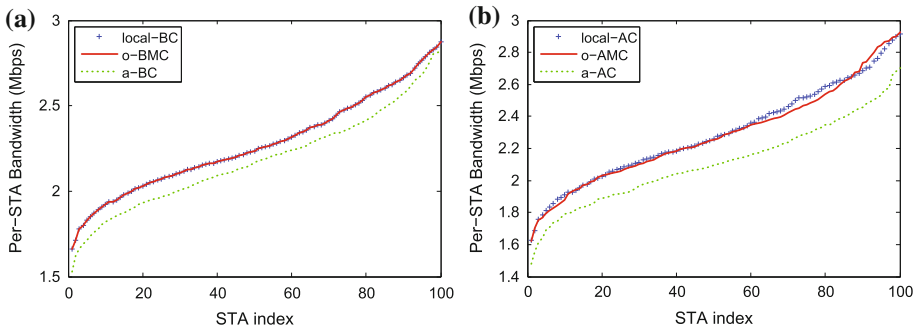


Fig. 7 Performance of the clique modeling methods. **a** Backhaul clique, **b** Access clique

5.3 Performance of JCBA

In the JCBA algorithm, the channel assignment is done by the JCaBa algorithm and the association control is done by the oAC algorithm. We compare the bandwidth allocation results of the algorithms in JCBA against other state-of-the-art channel assignment and association control schemes.

We compare the following channel assignment schemes:

- *VC* vertex colouring algorithm DSATUR [20] that is introduced in Sect. 2.
- *LCCS* least congested channel search where each MAP searches for the channel with the fewest number of STAs. If the association is changed due to some channel changes, repeat the process of LCCS and association control until reaching convergence.
- *JCaBa* joint channel assignment and bandwidth allocation algorithm that is used in the first step of JCBA.

We compare the following association control schemes:

- *SS* strongest signal, i.e. a STA associates with the MAP from which the received signal strength is the highest.
- *CL* cross-layer association metric [3]. The total association cost is a weighted sum of the access cost and the backhaul cost, which reflects the estimated amount of time consumed by a successful end-to-end packet transmission.
- *oAC* optimization-based association control algorithm that is used in the second step of JCBA.

Figure 8 depicts the per-STA bandwidth performance of six combinations of channel assignment and association control schemes under uniform user topology and hotspot user topology; Table 3 gives the corresponding numerical results of the aggregate throughput and Jain's fairness index. Under the same channel assignment done by the VC algorithm, we compare the performance of the association control schemes. The performance of CL is slightly better than that of SS, because CL considers not only the access network condition but also the backhaul condition. As a result, CL has more STAs associated with the good-backhaul MAPs and makes better use of the network resource. However, due to the nature of heuristic algorithms, CL has no optimization attempt and it has no consideration in the utility objective when making association decision. Therefore, it is as expected that CL significantly underperforms oAC in terms of both throughput and fairness. Under the hotspot topology, the bandwidth allocation in SS and CL are very unfair because too many STAs associate with the hotspot MAPs and the STAs associated with the non-hotspot MAPs are allocated with excessive bandwidth. In contrast, oAC is able to achieve very fair user bandwidth allocation no matter what the user topology is.

Using oAC as the association control algorithm, we compare the performance of the channel assignment schemes. In Fig. 8 and Table 3, it is clear that JCaBa achieves the best performance in the user bandwidth as well as the aggregate throughput. JCaBa significantly outperforms VC. VC equally weighs each MAP regardless of the MAP load. In contrast, in JCaBa, the interference experienced by the good-backhaul MAPs contributes more to the total interference, and it is minimized with priority. As JCaBa increases the capacity of the good-backhaul MAPs, more STAs can associate with these MAPs, which further improves the network resource utilization efficiency. Like JCaBa, LCCS also outperforms VC. The reason is that, by LCCS, the poor-backhaul MAPs will avoid the channels of the heavily-loaded good-backhaul MAPs, resulting in less interference at the good-backhaul MAPs and higher network capacity. JCaBa's performance is better than LCCS's because JCaBa uses a

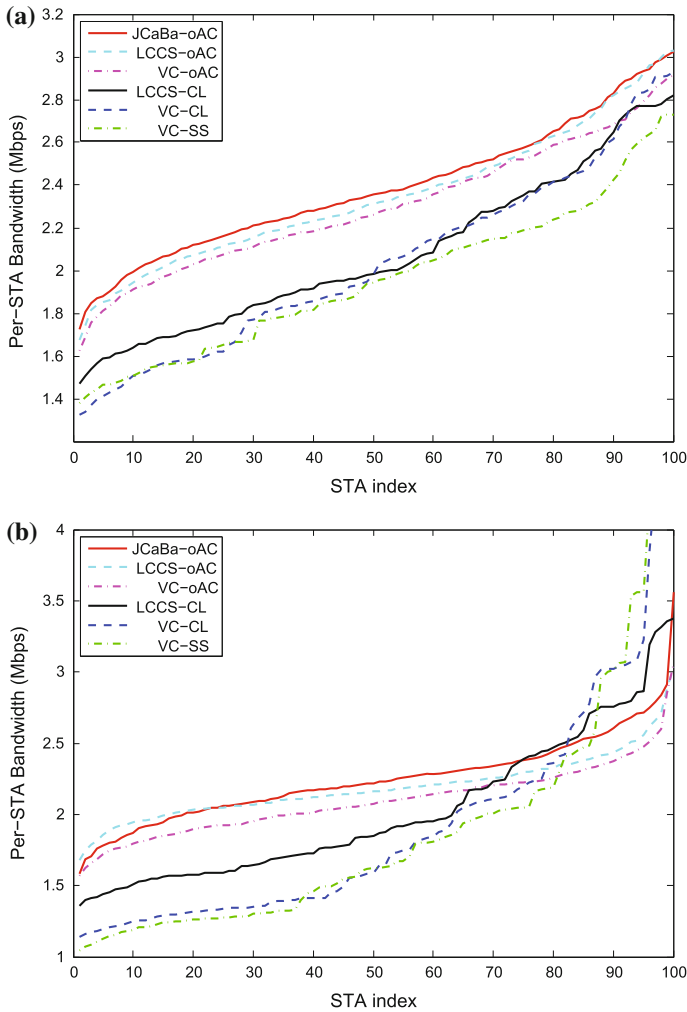


Fig. 8 Performance of the channel assignment and association control schemes. **a** Uniform topology, **b** Hotspot topology

Table 3 Aggregate throughput and fairness index of the channel assignment and association control schemes

	VC-SS	VC-CL	LCCS-CL	VC-oAC	LCCS-oAC	JCaBa-oAC
<i>Uniform</i>						
Throughput	195.70	203.45	207.69	229.43	234.61	238.40
Fairness	0.969	0.958	0.970	0.983	0.982	0.984
<i>Hotspot</i>						
Throughput	190.01	196.03	202.02	209.38	218.81	224.31
Fairness	0.815	0.811	0.940	0.986	0.990	0.982

more comprehensive load-weighted interference metric and bonds with the optimal association control more closely.

Comparing the channel assignment and association control schemes in Fig. 8, it is noticed that the performance improvement of oAC over SS/CL is much more significant than that of JCaBa over VC/LCCS. In other words, the performance improvement of the JCaBa algorithm is mainly contributed by the association control algorithm, rather than the channel assignment algorithm. This finding is consistent with that in [16] for WLANs. That is one of the reasons why oAC should be conducted more frequently than JCaBa, which is discussed in Sect. 4.3.

As it is NP-hard to find the optimal channel assignment for comparison, we measure the performance improvement of JCaBa by varying the number of the non-overlapping channels available in the access networks, and the results are given in Table 4. It is interesting to notice that, when the number of channels is 4 or 5, the throughput increment of JCaBa over VC almost equals the throughput increment that would be gained by adding one more channel to VC, i.e. $JCaBa-4CH : VC-5CH = 238.4 : 239.7$ and $JCaBa-5CH : VC-6CH = 249.1 : 249.7$. In other words, the channel utilization efficiency of JCaBa is about 20–25% higher than that of VC.

In Fig. 9, we compare the aggregate throughput of the 4 channel assignment algorithms as the number of non-overlapping channels increases from 3 to 10. In the Random scheme, we select the best channel assignment out of 10 random channel assignments. We can see that the performance improvement of our JCaBa algorithm over the Random scheme enlarges as the number of channels increases, which demonstrates the effectiveness of our method in reducing the interference at the good-backhaul MAPs with priority. The performance advantage of the JCaBa algorithm over the VC algorithm shrinks as the number of channels increases. That is because when the number of non-overlapping channels is large enough, the VC algorithm will be able to find channel assignments such that all adjacent MAPs are assigned with different channels and inter-cell interference can be eliminated. However, in reality, as the channel bandwidth becomes larger and larger, and more and more APs/MAPs are deployed, it is harder and harder to find so many non-overlapping channels. The advantage of JCaBa over LCCS is quite consistent as both of them take the MAP load into consideration.

The JCaBa results presented are based on the best of 10 runs of the JCaBa algorithm with different random initial channel assignments. In each run, when the number of channels is 4, 3.97 while-loops are conducted on average, with a standard deviation of 1.22. In each while-loop, $|M|$, which is 20 in our simulation, FracP convex problems are solved. So the JCaBa algorithm terminates within a few rounds of convex problem solving. Fig. 10 depicts the average number of while-loops conducted in the JCaBa algorithm versus the number of non-overlapping channels in the network. It is clear that when there are more non-overlapping channels, a fewer number of while-loops are conducted before

Table 4 Throughput performance of the channel assignment schemes

	3-CH	4-CH	5-CH	6-CH
VC	214.20	229.43	239.76	249.76
LCCS	216.88	234.39	245.23	252.77
JCaBa	221.81	238.40	249.12	258.59

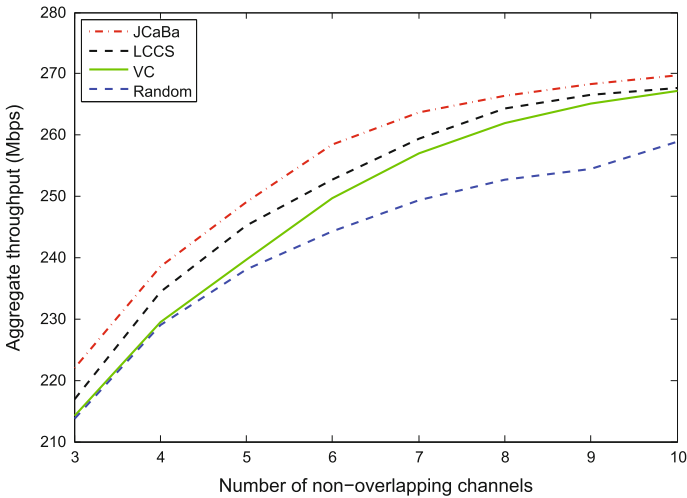


Fig. 9 Performance of the channel assignment algorithms

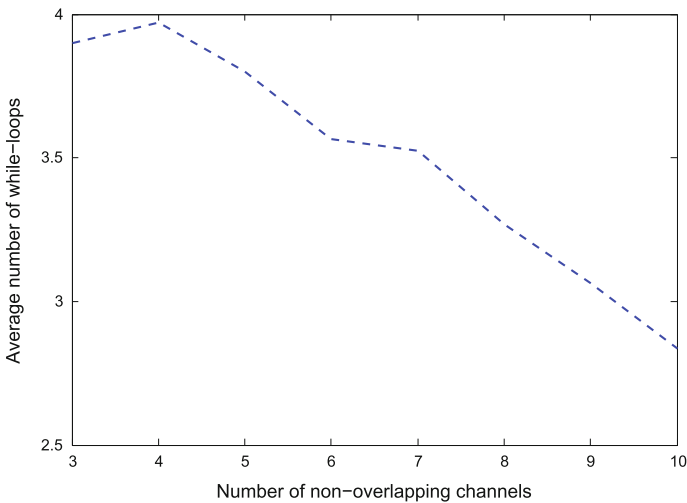


Fig. 10 Number of while-loops conducted

convergence. That demonstrates that the JCaBa algorithm converges faster in a network where a MAP’s channel change has less impact on the other MAPs as we have discussed in Sect. 4.2.

Besides the simulation results for the networks of 20 MAPs and 100 STAs presented above, we have also conducted simulation for networks of higher node density, where 40 MAPs and 200 STAs are randomly located in a square field of the same size as above. The numerical results are given in Table 5. In the higher density networks, all the schemes achieve better performance, and JCaBa outperforms the other schemes more significantly. That can be explained from two aspects. Firstly, the link rates are higher due to shorter inter-node distances. The second is more important that with more MAPs available in the

Table 5 Performance for the networks of higher node density

	VC-SS	VC-CL	LCCS-CL	VC-oAC	LCCS-oAC	JCaBa-oAC
<i>Uniform</i>						
Throughput	225.09	244.64	245.38	288.40	291.68	295.64
Fairness	0.983	0.981	0.981	0.988	0.986	0.985
<i>Hotspot</i>						
Throughput	232.80	239.21	247.98	270.20	277.04	282.02
Fairness	0.923	0.837	0.975	0.992	0.991	0.991

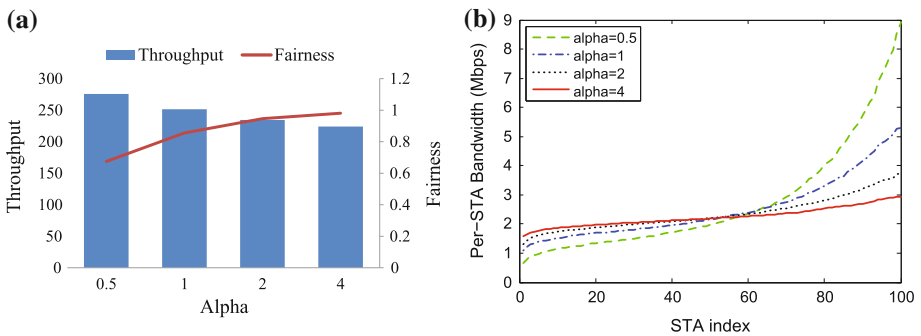


Fig. 11 Performance of UBa with different α value. **a** Aggregate throughput and fairness index, **b** Per-STA bandwidth

vicinity, a STA has more opportunities to associate with a good-backhaul MAP, which can be better utilized by oAC to find a better association. However, the performance improvement of JCaBa over VC/LCCS is not as significant as before, which can also be explained from two aspects. Firstly, due to the high node density, the interference at the good-backhaul MAPs cannot be eliminated or significantly reduced no matter how effective a channel assignment scheme is. Secondly, by oAC, a lot of STAs are already associated with the good-backhaul MAPs; so even though JCaBa can reduce the interference at the good-backhaul MAPs, there would not be many more STAs switching to associate with these MAPs.

Finally, we take a look at the performance of the utility-fair bandwidth allocation algorithm UBa. Figure 11a shows the aggregate throughput and Jain’s fairness index results when the fairness control parameter α is varied from 0.5 to 4. It is clear that as α increases, the aggregate throughput decreases and the fairness index increases. Figure 11b depicts the per-STA bandwidth results. It is obvious that with a larger α , the line is flatter, which indicates a fairer bandwidth allocation. With a smaller α , the line is steeper and the area below the line is larger, which indicates a less fair bandwidth allocation and higher aggregate throughput.

Note that the fairness index here reflects the fairness in terms of user bandwidth. There are also other definitions of fairness, such as the fairness in user transmission time. A proper α value should be determined by the network designer/operator, so that the desired fairness can be enforced. As α becomes larger and larger, we get better and better fairness in user bandwidth at the expense of lower and lower network resource utilization

efficiency. It is interesting to notice that in Fig. 11b, as α increases, the bandwidth of the low-bandwidth STAs on the left, which is about 60% of the entire STAs, slightly decreases, while the bandwidth of the high-bandwidth STAs on the right, which is the rest 40% of the STAs, dramatically increases. So it might be a good idea to select a smaller α .

6 Conclusion

In this paper, we have proposed a network resource management framework, named JCBA, for WMNs, which jointly considers channel assignment, association control, and bandwidth allocation. The JCBA framework is composed of three components: a utility-based bandwidth allocation algorithm that is flexible in adjusting the trade-off between resource utilization efficiency and user fairness in bandwidth; a channel assignment algorithm that can effectively increase the network capacity by reducing the interference at the good-backhaul MAPs; and an optimization-based association control algorithm that finds approximately optimal association solutions such that the network capacity can be further improved by letting more STAs associate with the good-backhaul MAPs. We have demonstrated the superior performance of the proposed algorithms through simulations with various network topologies and conditions.

References

1. Akyildiz, I. F., Wang, X., & Wang, W. (2005). Wireless mesh networks: a survey. *Computer Networks*, 47(4), 445–487.
2. Gamberoza, V., Sadeghi, B., & Knightly, E.W. (2004). End-to-end performance and fairness in multihop wireless backhaul networks. In *Proceedings of ACM MobiCom* (pp. 287–301).
3. Yu, J., & Wong, W.C. (2012). Network resource aware association control in wireless mesh networks. In *Proceedings of IEEE ICCS* (pp. 368–372).
4. Bejerano, Y., Han, S.-J., & Li, L. (2007). Fairness and load balancing in wireless lans using association control. *IEEE/ACM Transactions on Networking*, 15(3), 560–573.
5. Li, L., Pal, M., & Yang, Y.R. (Apr 2008). Proportional fairness in multi-rate wireless lans. In *Proceedings of IEEE INFOCOM* (pp. 1004–1012).
6. Athanasiou, G., Korakis, T., Ercetin, O., & Tassiulas, L. (2009). A cross-layer framework for association control in wireless mesh networks. *IEEE Transactions on Mobile Computing*, 8(1), 65–80.
7. Luo, L., Raychaudhuri, D., Liu, H., Wu, M., & Li, D. (Nov 2008). Joint association, routing and bandwidth allocation for wireless mesh networks. In *Proceedings of IEEE GLOBECOM* (pp. 1–6).
8. Yu, J., & Wong, W. C. (2015). Optimal association in wireless mesh networks. *IEEE Transactions on Vehicular Technology*, 64(5), 2084–2096.
9. Chiochan, S., Hossain, E., & Diamond, J. (2010). Channel assignment schemes for infrastructure-based 802.11 w lans: A survey. *IEEE Communications Surveys and Tutorials*, 12(1), 124–136.
10. Radunovic, B., & Boudec, J.-Y. L. (2004). Rate performance objectives of multihop wireless networks. *IEEE Transactions on Mobile Computing*, 3(4), 334–349.
11. Heusse, M., Rousseau, F., Berger-Sabbatel, G., & Duda, A. (2003). Performance anomaly of 802.11b. *Proceedings of the IEEE INFOCOM*, 2, 836–843.
12. Kelly, F. P. (1997). Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8(1), 33–37.
13. Babu, A.V., & Jacob, L. (June 2005). Performance analysis of ieee 802.11 multirate w lans: time based fairness vs throughput based fairness. In *Proceedings of IEEE WIRELESSCOM* (pp. 203–208).
14. Li, W., Wang, S., Cui, Y., Cheng, X., Xin, R., & Al-Rodhaan, M. A., et al. (2014). AP association for proportional fairness in multirate WLANs. *IEEE/ACM Transactions on Networking*, 22(1), 191–202.
15. Hung, K., & Bensaou, B. (2014). Throughput optimization in wireless local networks with inter-AP interference via a joint-association control, rate control, and contention resolution. *Ad Hoc Networks*, 12, 115–129.

16. Zhang, J., Wang, J.M., & Bensaou, B. (July 2013). Optimal resource sharing in multi-cell wireless LANs: joint channel assignment, association control, airtime sharing and contention resolution. In *Proceedings of 9th international wireless communications and mobile computing conference (IWCMC)* (pp. 317–322).
17. Alim Al Islam, A. B. M., Islam, M. J., Nurain, N. & Raghunathan, V. (2016). Channel assignment techniques for multi-radio wireless mesh networks: A survey. *IEEE Communications Surveys Tutorials*, 18(2), 988–1017.
18. Achanta, M. (April 2006). Method and apparatus for least congested channel scan for wireless access points. US Patent No. 20060072602.
19. Kauffmann, B., Baccelli, F., Chaintreau, A., Mhatre, V., Papagiannaki, K., & Diot, C. (May 2007). Measurement-based self organization of interfering 802.11 wireless access networks. In *Proceedings of IEEE INFOCOM* (pp. 1451–1459).
20. Mahonen, P., Riihijarvi, J., & Petrova, M. (Sept 2004). Automatic channel allocation for small wireless local area networks using graph colouring algorithm approach. In *Proceedings of IEEE PIMRC* (pp. 536–539).
21. Mishra, A., Brik, V., Banerjee, S., Srinivasan, A., & Arbaugh, W. (April 2006). A client-driven approach for channel management in wireless lans. In *Proceedings of IEEE INFOCOM* (pp. 1–12).
22. (Sept 2011) IEEE 802.11 standard: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 10: Mesh networking (pp. 1–372)
23. Bron, C., & Kerbosch, J. (1973). Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16(9), 575–577.
24. Mo, J., & Walrand, J. (2000). Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on Networking*, 8(5), 556–567.
25. Broustis, I., Papagiannaki, K., Krishnamurthy, S. V., Faloutsos, M., & Mhatre, V. P. (2010). Measurement-driven guidelines for 802.11 wlan design. *IEEE/ACM Transactions on Networking*, 18(3), 722–735.
26. Cisco wireless mesh access points, design and deployment guide release7-5. [Online]. Available: <http://www.cisco.com/en/US/docs/wireless/technology/mesh/7.5/design/guide/mesh75.html>
27. R. Jain, Chiu, D., & Hawe, W. (September 1984). A quantitative measure of fairness and discrimination for resource allocation in shared computer system. DEC Research Report TR-301.



Jinqiang Yu received the B.Eng. (1st class Honours) and Ph.D. degrees in Electrical Engineering from National University of Singapore, Singapore, in 2008 and 2015, respectively. He was a wireless engineer and software engineer at Huawei Technologies Singapore and IBM Singapore, respectively, from 2008 to 2010. He is a Scientist in the Institute for Infocomm Research, A*STAR, Singapore. His research interests include wireless networking, wireless mesh networks, resource allocation and Intelligent Transportation Systems.



Wai-Choong Wong received the B.Sc. (1st class Honours) and Ph.D. degrees in Electronic and Electrical Engineering from Loughborough University, UK, in 1976 and 1980, respectively. He is Professor in the Department of Electrical and Computer Engineering, National University of Singapore (NUS). He is currently holding the position of Deputy Director (Strategic Development) at the Interactive and Digital Media Institute (IDMI) in NUS. He was previously Executive Director of the Institute for Infocomm Research (I2R) from Nov 2002 to Nov 2006. Since joining NUS in 1983, he served in various positions at the department, faculty, and university levels, including Head of the Department of Electrical and Computer Engineering from Jan 2008 to Oct 2009, Director of the NUS Computer Centre from Jul 2000 to Nov 2002, and Director of the Centre for Instructional Technology from Jan 1998 to Jun 2000. Prior to joining NUS in 1983, he was a Member of Technical Staff at AT&T Bell Laboratories, Crawford Hill Lab, NJ, USA, from 1980 to 1983. His research interests include wireless networks and systems, multimedia networks, and source matched transmission techniques with over 200 publications and 4 patents in these areas. He is co-author of the book on “Source-Matched Mobile Communications”. He received the IEEE Marconi Premium Award in 1989, NUS Teaching Award (1989), IEEE Millennium Award in 2000, the innovator Awards 2000, Open Category, and Best Paper Award at the IEEE International Conference on Multimedia and Expo (ICME) 2006. He is a senior member of the IEEE.

networks and systems, multimedia networks, and source matched transmission techniques with over 200 publications and 4 patents in these areas. He is co-author of the book on “Source-Matched Mobile Communications”. He received the IEEE Marconi Premium Award in 1989, NUS Teaching Award (1989), IEEE Millennium Award in 2000, the innovator Awards 2000, Open Category, and Best Paper Award at the IEEE International Conference on Multimedia and Expo (ICME) 2006. He is a senior member of the IEEE.